Joseph Finkelstein
Robert Moskovitch
Enea Parimbelli (Eds.)

LNAI 14844

Artificial Intelligence in Medicine

22nd International Conference, AIME 2024
Salt Lake City, UT, USA, July 9–12, 2024
Proceedings, Part I

Part I

AIME 2024
22nd International Conference on Artificial Intelligence in Medicine
Salt Lake City, Utah, USA, July 9-12
Hosted by the University of Utah

Springer

The series Lecture Notes in Artificial Intelligence (LNAI) was established in 1988 as a topical subseries of LNCS devoted to artificial intelligence.

The series publishes state-of-the-art research results at a high level. As with the LNCS mother series, the mission of the series is to serve the international R & D community by providing an invaluable service, mainly focused on the publication of conference and workshop proceedings and postproceedings.

Joseph Finkelstein · Robert Moskovitch ·
Enea Parimbelli
Editors

Artificial Intelligence in Medicine

22nd International Conference, AIME 2024
Salt Lake City, UT, USA, July 9–12, 2024
Proceedings, Part I

Springer

Editors

Joseph Finkelstein
University of Utah
Salt Lake City, UT, USA

Robert Moskovitch
Ben-Gurion University of the Negev
Beer Sheva, Israel

Enea Parimbelli
University of Pavia
Pavia, Italy

ISSN 0302-9743 ISSN 1611-3349 (electronic)
Lecture Notes in Artificial Intelligence
ISBN 978-3-031-66537-0 ISBN 978-3-031-66538-7 (eBook)
https://doi.org/10.1007/978-3-031-66538-7

LNCS Sublibrary: SL7 – Artificial Intelligence

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

The Society for Artificial Intelligence in Medicine (AIME) was established in 1986 following a very successful workshop held in Pavia, Italy, the year before. The principal aims of AIME are to foster fundamental and applied research in the application of Artificial Intelligence (AI) techniques to medical care and medical research, and to provide a forum at conferences for discussing any progress made. The main activity of the society thus far has been the organization of a series of international conferences, held in Marseille, France (1987), London, UK (1989), Maastricht, Netherlands (1991), Munich, Germany (1993), Pavia, Italy (1995), Grenoble, France (1997), Aalborg, Denmark (1999), Cascais, Portugal (2001), Protaras, Cyprus (2003), Aberdeen, UK (2005), Amsterdam, Netherlands (2007), Verona, Italy (2009), Bled, Slovenia (2011), Murcia, Spain (2013), Pavia, Italy (2015), Vienna, Austria (2017), Poznan, Poland (2019), Minneapolis, USA (2020), Porto, Portugal (2021), Halifax, Canada (2022), and Portoroz, Slovenia (2023).

The AIME 2024 goals were to present and consolidate the international state of the art of AI in biomedical research from the perspectives of theory, methodology, systems, and applications. The conference included three invited keynotes, presentations of full, short, and demonstration papers, tutorials, workshops, a doctoral consortium, and posters. Overall, AIME 2024 received 335 submissions from which 54 full papers and 22 short papers were chosen for oral presentation at the main conference and for publication in two volumes of conference proceedings. Submissions came from authors in 46 countries, including submissions from Europe, North and South America, Africa, Australia, and Asia.

All papers were carefully peer-reviewed by experts from the Program Committee, and subsequently by members of the Senior Program Committee. Each submission was single-blindly reviewed in most cases by three reviewers, and all papers by at least two reviewers. The reviewers judged the overall quality of the submitted papers, together with their relevance to the AIME conference, originality, impact, technical correctness, methodology, scholarship, and quality of presentation. In addition, the reviewers provided detailed written comments on each paper and stated their confidence in the subject area. One Senior Program Committee member was assigned to each paper, who wrote a meta-review and provided a recommendation to the Organizing Committee. A dedicated expert committee comprising the AIME 2024 general chairs, Joseph Finkelstein and Robert Moskovitch, the Scientific Program Committee chair, Enea Parimbelli, and the doctoral consortium chair, Gregor Štiglic, made the final decisions regarding the AIME 2024 conference program. This process was carried out with virtual meetings starting in February 2024. Each long paper was presented in a 20-minute oral presentation during the conference. Each regular short and demonstration paper was presented in a 10-minute presentation, a poster or a demonstration session. The main conference poster session hosted over 70 poster presentations.

The papers were organized according to their topics in the following main themes: (1) Predictive modelling and disease risk prediction; (2) Natural language processing; (3) Bioinformatics and omics; (4) Wearable devices, sensors, and robotics; (5) Medical image analysis; (6) Data integration and multimodal analysis; (7) Explainable AI.

AIME 2024 had the privilege of hosting three invited keynote speakers: Nicholas Tatonetti, Vice Chair of Operations in the Department of Computational Biomedicine and Associate Director of Computational Oncology at Cedars-Sinai Medical Center, Los Angeles, USA, giving the keynote entitled “AI-driven biomedical discoveries using real-world data,” Hayit Greenspan, Co-director of the Artificial Intelligence and Emerging Technologies in Medicine PhD concentration at the Icahn School of Medicine at Mount Sinai, New York, USA, who discussed “AI in Medical Imaging – Steps towards supporting Detection and Monitoring of Disease,” and Yuan Luo, Director, Institute for Artificial Intelligence in Medicine, Northwestern University, Chicago, USA, who presented “Translational Science: Breakthroughs and Innovations in the new Age of AI.”

The doctoral consortium received 11 PhD proposals that were peer reviewed. AIME 2024 provided an opportunity for six of these PhD students to discuss their research goals, proposed methods, and preliminary results in an oral presentation and for four students to present a poster. A scientific panel, consisting of experienced researchers in the field, provided constructive feedback to the students in an informal atmosphere. The doctoral consortium was chaired by Dr. Gregor Štiglic (University of Maribor, Slovenia).

AIME 2024 invited researchers to submit proposals for workshops and tutorials. Twelve workshops were selected by the Organization Committee. These workshops were: (1) Advances in Generating Real-World Evidence from Real-World Data Using Artificial Intelligence (RWE-AI), (2) AI Applications in Telemedicine and Digital Health (TeleHealth-AI), (3) Artificial Intelligence in Oncology Workshop, (4) Third International Workshop on Artificial Intelligence in Nursing (AINurse-24), (5) AI for Drug Discovery: Development in Pharmaceuticals, Academia, or Jointly in Collaborations, (6) AI for Reliable and Equitable Real World Evidence Generation in Medicine, (7) AI for Primary Care: Electronic Scribes and More, (8) AI and Precision Medicine: Innovations and Applications, (9) Implementing AI in Healthcare: Bridging the Gap Between Technical, Cognitive, and Sociotechnical Perspectives, (10) AI Applications in Public Health and Social Services, (11) Mitigating AI Risk through Ethical Data Science and (12) Veteran Health Administration: AI for Clinical Informatics Solutions. In addition to the workshops, two interactive half-day tutorials were selected: (1) Process Mining for Healthcare: From Theory to Practice with pMineR, and (2) Integrating Geography with AI to Improve Decision Support in Public Health.

We would like to thank everyone who contributed to the AIME 2024 conference. First, we would like to thank the authors of the submitted papers and posters and the members of the Program Committee who helped with conducting a successful review. Thank you to the Senior Program Committee for writing meta-reviews and the members of the Senior Advisory Board for guidance. Thanks are also due to the invited speakers as well as to the organizers of the tutorials, the workshops, and the doctoral consortium panel. Many thanks go to the local Organizing Committee members, who managed all the work making this conference possible: Yves Lussier, Penny Atkins, Yue Zhang, Fatemeh Shah-Mohammadi, Aref Smiley, Te-Yi Tsai, and C. Mahony Reategui-Rivera. We would like to thank Springer for sponsoring the conference, the AIME society for sponsoring awards and the student travel grants, and the University of Utah, Department of Biomedical Informatics, for all the support in organizing it. Finally, we thank the Springer team for helping us in the final preparation of this LNAI book.

May 2024

Joseph Finkelstein
Robert Moskovitch
Enea Parimbelli

Organization

General Chairs

Joseph Finkelstein	University of Utah, USA
Robert Moskovitch	Ben-Gurion University of the Negev, Israel

Program Committee Chairs

Enea Parimbelli (Scientific Program)	University of Pavia, Italy
Xiaoqian Jiang (Scientific Program)	University of Texas at Houston, USA
Marina Sirota (Scientific Program)	University of California at San Francisco, USA
Yves Lussier (Senior Advisory Board)	University of Utah, USA
Panagiotis Papapetrou (Workshop)	Stockholm University, Sweden
Fadia Shaya (Workshop)	University of Maryland, USA
Aref Smiley (Workshop)	University of Utah, USA
Myra Spiliopoulou (Tutorial)	Otto von Guericke University Magdeburg, Germany
Francesca Vitali (Tutorial)	University of Arizona, USA
Fatemeh Shah-Mohammadi (Tutorial)	University of Utah, USA
Gregor Štiglic (Doctoral Consortium)	University of Maribor, Slovenia
Colleen Kenost (Publicity)	University of Utah, USA

Local Organizing Committee

Yves Lussier (Co-chair)	University of Utah, USA
Joseph Finkelstein (Co-chair)	University of Utah, USA
Penny Atkins	University of Utah, USA
Yue Zhang	University of Utah, USA
Fatemeh Shah-Mohammadi	University of Utah, USA

Aref Smiley	University of Utah, USA
Te-Yi Tsai	University of Utah, USA
C. Mahony Reategui-Rivera	University of Utah, USA

Senior Advisory Board

Aaron Quinlan	University of Utah, USA
Aik Choon Tan	University of Utah, USA
Atul Butte	University of California at San Francisco, USA
Brett Beaulieu-Jones	University of Chicago, USA
Carlo Combi	University of Verona, Italy
David Classen	University of Utah, USA
Dina Demner-Fushman	National Library of Medicine, USA
Fadia Shaya	University of Maryland, USA
Frank van Harmelen	Vrije Universiteit Amsterdam, Netherlands
Hua Xu	Yale University, USA
James M. Hotaling	University of Utah, USA
Jason Moore	Cedars-Sinai Medical Center, USA
Jiaje Zhang	University of Texas Health Science Center at Houston, USA
John Holmes	University of Pennsylvania, USA
Ju Han Kim	Seoul National University, South Korea
Kun Huang	Indiana University, USA
Leonardo Rojas-Mezarina	Ministry of Health, Peru
Luca Giancarlo	University of Texas Health Science Center at Houston, USA
Lucila Ohno-Machado	Yale University, USA
Manish Kohli	University of Utah, USA
Manish Parashar	University of Utah, USA
Mark Craven	University of Wisconsin-Madison, USA
Mark Musen	Stanford University, USA
Martin Michalowski	University of Minnesota, USA
Mattia Prosperi	University of Florida, USA
Mike Kirby	University of Utah, USA
Milos Hauskrecht	University of Pittsburgh, USA
Mor Peleg	University of Haifa, Israel
Nicholas Tatonetti	Cedars-Sinai Medical Center, USA
Niels Peek	University of Manchester, UK
Philip Payne	Washington University in St. Louis, USA
Riccardo Bellazzi	University of Pavia, Italy
Robert Moskovitch	Ben-Gurion University of the Negev, Israel

Serena Jingchuan Guo	University of Florida, USA
Syed Sibte Raza Abidi	Dalhousie University, Canada
Tomohiro Sawa	Teikyo University, Japan
Xinxin (Katie) Zhu	Yale University, USA
Yu-Chuan (Jack) Li	Taipei Medical University, Taiwan
Zhongming Zhao	University of Texas Health Science Center at Houston, USA

Senior Program Committee

Abu Mosa	University of Missouri, USA
Adiebonye Jumbo	SUNY Downstate Health Sciences University, USA
Allan Tucker	Brunel University, UK
Annette ten Teije	Vrije Universiteit Amsterdam, Netherlands
Arianna Dagliati	University of Pavia, Italy
Barbara Oliboni	University of Verona, Italy
Blaz Zupan	University of Ljubljana, Slovenia
Brad Malin	Vanderbilt University, USA
Carlo Combi	Università degli Studi di Verona, Italy
Christopher Tignanelli	University of Minnesota, USA
Craig Kuziemsky	MacEwan University, Canada
Daniel Capurro	University of Melbourne, Australia
Daniel Otzoy	Central American Health Informatics Network, Guatemala
Ece Uzun	Brown University, USA
Elpida Keravnou-Papailiou	University of Cyprus, Cyprus
Gregor Štiglic	University of Maribor, Slovenia
Hamilton Baker	Medical University of South Carolina, USA
Hua Xu	Yale University, USA
Janna Hastings	University of Zurich, Switzerland
John Holmes	University of Pennsylvania, USA
Jonathan Golob	University of Michigan, USA
Jose M. Juarez	Universidad de Murcia, Spain
Kaifu Chen	Harvard University, USA
Kenrick Cato	University of Pennsylvania, USA
Kun Huang	Regenstrief Institute, USA
Li Shen	University of Pennsylvania, USA
Luca Giancarlo	University of Texas Health Science Center at Houston, USA
Lucia Sacchi	University of Pavia, Italy

Mar Marcos	Universitat Jaume I, Spain
Margarita Sordo	Harvard University, USA
Mark Parkulo	Mayo Clinic, USA
Martin Michalowski	University of Minnesota, USA
Maryam Zolnoori	Columbia University, USA
Michael E. Matheny	Vanderbilt University, USA
Milos Hauskrecht	University of Pittsburgh, USA
Mor Peleg	University of Haifa, Israel
Nada Lavrač	Jožef Stefan Institute, Slovenia
Naleef Fareed	Ohio State University, USA
Nan Liu	Duke-NUS Medical School, Singapore
Niels Peek	University of Manchester, UK
Pedro Henriques Abreu	University of Coimbra, Portugal
Peter Lucas	University of Twente, Netherlands
Rui Zhang	University of Minnesota, USA
Sarath Chandra Janga	Indiana University – Purdue University Indianapolis, USA
Silvana Quaglini	University of Pavia, Italy
Stefania Montani	Università del Piemonte Orientale, Italy
Steve Johnson	University of Minnesota, USA
Syed Sibte Raza Abidi	Dalhousie University, Canada
Szymon Wilk	Poznan University of Technology, Poland
Tianxi Cai	Harvard University, USA
Victor Maojo	Universidad Politécnica de Madrid, Spain
Wenyu Song	Brigham and Women's Hospital, USA
Yanshan Wang	University of Pittsburgh, USA
Young Ji Lee	University of Pittsburgh, USA
Yuval Shahar	Ben-Gurion University of the Negev, Israel

Program Committee

Abdolrahman Peimankar	University of Southern Denmark, Denmark
Abeed Sarker	Emory University, USA
Ahmad P. Tafti	University of Pittsburgh, USA
Aida Kamisalic	University of Maribor, Slovenia
Aize Cao	Meharry Medical College, USA
Aleksander Sadikov	University of Ljubljana, Slovenia
Alessio Bottrighi	Università del Piemonte Orientale, Italy
Alfredo Vellido	Universitat Politècnica de Catalunya, Spain
Allan Tucker	Brunel University, UK
Amin Zollanvari	Nazarbayev University, Kazakhstan

Ana Maria Mendonça	Institute for Systems and Computer Engineering, Technology and Science, Portugal
André Fujita	University of São Paulo, Brazil
Anna Fabijańska	Technical University of Lodz, Poland
Annette ten Teije	Vrije Universiteit Amsterdam, Netherlands
Ardalan Naseri	University of Texas Health Science Center at Houston, USA
Aref Smiley	University of Utah, USA
Arianna Dagliati	University of Pavia, Italy
Arif Harmanci	University of Texas Health Science Center at Houston, USA
Arjen Hommersom	Open University of the Netherlands, Netherlands
Arvind Rao	University of Michigan, USA
Barbara Di Camillo	University of Padova, Italy
Baris Suzek	Muğla Sitki Koçman University, Turkey
Beatriz López	University of Girona, Spain
Begoña Martinez-Salvador	Universitat Jaume I, Spain
Bernard Chen	University of Central Arkansas, USA
Bernardo Canovas Segura	Universidad de Murcia, Spain
Bertha Guijarro-Berdiñas	Universidade da Coruña, Spain
Bhasuran Balu	Florida State University, USA
Bibek Paudel	Stanford University, USA
Binh Vu	Research Institute for Telecommunication and Cooperation, Germany
Brad Malin	Vanderbilt University, USA
Brian Locke	University of Utah, USA
Carlo Combi	Università degli Studi di Verona, Italy
Carmela Comito	Institute for High Performance Computing and Networking, Italy
Carson Leung	University of Manitoba, Canada
Chao Cheng	Baylor College of Medicine, USA
Charles Kahn	University of Pennsylvania, USA
Chuming Chen	University of Delaware, USA
Chung-Sheng Li	PwC Information, UK
Cihan Bilge Kayasandık	Istanbul Medipol University, Turkey
Cuncong Zhong	University of Kansas, USA
Daniela Dauria	Free University of Bozen-Bolzano, Italy
Daniela Ferreira-Santos	University of Porto, Portugal
Daniele Pala	University of Pavia, Italy
Denis Newman-Griffis	University of Sheffield, UK
Dvir Aran	Technion – Israel Institute of Technology, Israel
Eleni Kaldoudi	Democritus University of Thrace, Greece

Eneida Mendonca	University of Cincinnati, USA
Erez Shalom	Ben-Gurion University of the Negev, Israel
Fangxiang Wu	University of Saskatchewan, Canada
Fatemeh Shah-Mohammadi	University of Utah, USA
Fernando Luís Barroso Da Silva	University of São Paulo, Brazil
Filip Jagodzinski	Western Washington University, USA
Fleur Mougin	Université de Bordeaux, France
Floriano Scioscia	Polytechnic University of Bari, Italy
Gayo Diallo	Université de Bordeaux, France
Georg Dorffner	Medical University of Vienna, Austria
Giltae Song	Pusan National University, South Korea
Giorgio Leonardi	Università del Piemonte Orientale, Italy
Giorgio Vinciguerra	University of Pisa, Italy
Giovanna Rosone	University of Pisa, Italy
Giovanni Delnevo	University of Bologna, Italy
Giuseppe Agapito	University Magna Graecia of Catanzaro, Italy
Guo Yi	University of Florida, USA
Hajer Baazoui	CY Cergy Paris University, France
Haohan Wang	University of Illinois at Urbana-Champaign, USA
Haridimos Kondylakis	Institute of Computer Science, Greece
Henrik Boström	KTH Royal Institute of Technology, Sweden
Hossein Estiri	Harvard University, USA
Hu Huang	University of Minnesota, USA
Hua Xu	University of Texas Health Science Center at Houston, USA
Iacopo Vagliano	Amsterdam Universitair Medische Centra, Netherlands
Ioanna Miliou	Stockholm University, Sweden
Isabel Sassoon	Brunel University, UK
Isabelle Bichindaritz	State University of New York at Oswego, USA
Jaebum Kim	Konkuk University, South Korea
Jake Chen	University of Alabama at Birmingham, USA
Jan Egger	Graz University of Technology, Austria
Jędrzej Potoniec	Poznan University of Technology, Poland
Je-Keun Rhee	Soongsil University, South Korea
Jens Weber	University of Victoria, Canada
Jerzy Błaszczyński	Poznan University of Technology, Poland
Jesualdo Tomás Fernández-Breis	Universidad de Murcia, Spain
Jie Xu	University of Florida, USA
Jim Zheng	University of Texas Health Science Center at Houston, USA
Jingwen Yan	Indiana University, USA

Joao Setubal	University of São Paulo, Brazil
Joe Song	New Mexico State University, USA
John Holmes	University of Pennsylvania, USA
Jordan Wyrwa	Children's Hospital Colorado, USA
Joseph Romano	University of Pennsylvania, USA
Juexin Wang	Indiana University, USA
Junbai Wang	University of Oslo, Norway
K. S. M. Tozammel Hossain	University of North Texas, USA
Kaifu Chen	Harvard University, USA
Kenrick Cato	University of Pennsylvania, USA
Kenta Nakai	University of Tokyo, Japan
Kerstin Denecke	Bern University of Applied Sciences, Switzerland
Kun-Mao Chao	National Taiwan University, Taiwan
Kyu-Baek Hwang	Soongsil University, South Korea
Laura Moss	University of Aberdeen, UK
Leng Han	Indiana University – Purdue University Indianapolis, USA
Licong Cui	University of Texas Health Science Center at Houston, USA
Lina Sulieman	Vanderbilt University, USA
Liqing Zhang	Virginia Polytechnic Institute and State University, USA
Lisiane Pruinelli	University of Florida, USA
Logan Pierce	University of California at San Francisco, USA
Loris Nanni	University of Padua, Italy
Luca Denti	Università degli Studi di Milano-Bicocca, Italy
Luca Piovesan	Università del Piemonte Orientale, Italy
Luis Rueda	University of Windsor, Canada
Luke Rasmussen	Northwestern University, USA
Manuel Campos	Universidad de Murcia, Spain
Manuel Striani	Università del Piemonte Orientale, Italy
Mar Marcos	Universitat Jaume I, Spain
Marco Spruit	Leiden University, Netherlands
Marcos L. P. Bueno	Eindhoven University of Technology, Netherlands
Martin Chapman	King's College London, UK
Martin Michalowski	University of Minnesota, USA
Maryam Zolnoori	Columbia University, USA
Mei Liu	University of Florida, USA
Mengdi Huai	Iowa State University, USA
Michael Ignaz Schumacher	University of Applied Sciences Western Switzerland, Switzerland
Milos Hauskrecht	University of Pittsburgh, USA

Ming-Yuan Chih	University of Kentucky, USA
Mohamed Nounou	Texas A&M University at Qatar, Qatar
Mor Peleg	University of Haifa, Israel
Morihiro Hayashida	Matsue College, Japan
Nada Lavrac	Jožef Stefan Institute, Slovenia
Nadav Rappoprt	Ben-Gurion University of the Negev, Israel
Nadia Pisanti	University of Pisa, Italy
Nansu Zong	Mayo Clinic, USA
Natalia Grabar	Université de Lille, France
Neil Smalheiser	University of Illinois at Chicago, USA
Nevo Itzhak	Ben-Gurion University of the Negev, Israel
Nguyen Quoc Khanh Le	Taipei Medical University, Taiwan
Niels Peek	University of Manchester, UK
Nisha Puthiyedth	Thompson Rivers University, Canada
Ognjen Arandjelovic	University of St Andrews, Scotland
Panagiotis Papapetrou	Stockholm University, Sweden
Pedro Furtado	University of Coimbra, Portugal
Pedro Larranaga	University of Madrid, Spain
Pedro Pereira Rodrigues	University of Porto, Portugal
Pierre Zweigenbaum	Université Paris-Saclay, France
Primoz Kocbek	University of Maribor, Slovenia
Qianqian Song	University of Florida, USA
Rafał Jóźwiak	Warsaw University of Technology, Poland
Ravi Janardan	University of Minnesota, USA
Reda Alhajj	University of Calgary, Canada
Ricardo Cardoso Pereira	University of Coimbra, Portugal
Rita Casadio	University of Bologna, Italy
Ronald Piscotty	Oakland University, USA
Rui Yin	University of Florida, USA
Rui Zhang	University of Minnesota, USA
Ruisheng Wang	Harvard University, USA
Ruoyu Wang	University of Texas Health Science Center at Houston, USA
Ryan Urbanowicz	University of Pennsylvania, USA
Saeed Salem	Qatar University, Qatar
Samina Abidi	Dalhousie University, Canada
Sanja Avramovic	George Mason University, USA
Satya Sahoo	Case Western Reserve University, USA
Sejung Yang	Yonsei University, South Korea
Seungyoon Nam	Gachon University, South Korea
Shanmughavel Piramanayagam	Bharathiar University, India

Shubo Tian	National Center for Biotechnology Information, USA
Shu-Kay Ng	Griffith University, Australia
Silvana Quaglini	University of Pavia, Italy
Silvia Miksch	Vienna University of Technology, Austria
Simona E. Rombo	Università degli Studi di Palermo, Italy
Simone Ciccolella	University of Milano-Bicocca, Italy
Sing-Hoi Sze	Texas A&M University at Qatar, Qatar
Siru Liu	Vanderbilt University, USA
Soon Chun	City University of New York, USA
Stefania Montani	Università del Piemonte Orientale, Italy
Szymon Wilk	Poznan University of Technology, Poland
Terri Elizabeth Workman	George Washington University, USA
Tian Shubo	National Center for Biotechnology Information, USA
Tiffany Callahan	Columbia University, USA
Valerio Guarrasi	Università Campus Bio-Medico di Roma, Italy
Vasudevan Jagannathan	3M M*Modal, USA
Vipina K. Keloth	Yale University, USA
W. Jim Zheng	University of Texas Health Science Center at Houston, USA
Wenjun Lin	Algoma University, Canada
Xia Ning	Ohio State University, USA
Xiaobo Zhou	University of Texas Health Science Center at Houston, USA
Xiaoming Zeng	University of North Carolina at Chapel Hill, USA
Xiaoyi Raymond Gao	Ohio State University, USA
Xiayuan Huang	University of Wisconsin–Madison, USA
Xing He	University of Florida, USA
Xinghua Shi	Temple University, USA
Xingquan Zhu	Florida Atlantic University, USA
Xu Jie	University of Florida, USA
Xuan Guo	University of North Texas, USA
Yanshan Wang	University of Pittsburgh, USA
Yasir Tarabichi	MetroHealth System, USA
Yejin Kim	University of Texas Health Science Center at Houston, USA
Yi Guo	University of Florida, USA
Ying Li	Regeneron Pharmaceuticals, USA
Yiqing Shen	Johns Hopkins University, USA
Younghee Lee	University of Utah, USA
Yu Zhang	Lehigh University, USA

Yuan An	Drexel University, USA
Yuanxi Fu	University of Illinois at Urbana-Champaign, USA
Yue Zhang	University of Utah, USA
Zhao Li	University of Texas Health Science Center at Houston, USA
Zhe He	Florida State University, USA
Zhengxing Huang	Zhejiang University, China
Zhiyu Wan	Vanderbilt University Medical Center, USA

Contents – Part I

Predictive Modelling and Disease Risk Prediction

Natural Language Processing

Bioinformatics and Omics

Wearable Devices, Sensors, and Robotics

Contents – Part II

Medical Imaging Analysis

Data Integration and Multimodal Analysis

Explainable AI

Predictive Modelling and Disease Risk Prediction

Applying Gaussian Mixture Model for Clustering Analysis of Emergency Room Patients Based on Intubation Status

Po-Kuang Chen[1,2](✉), Shih-Hsien Sung[2](✉), and Ling Chen[3](✉)

[1] Kuang-Tien General Hospital, Taichung City, Taiwan
drkuang.md10@nycu.edu.tw
[2] Institute of Emergency and Critical Care Medicine, National Yang Ming Chiao Tung University, Hsinchu, Taiwan
[3] Institute of Hospital and Health Care Administration, National Yang Ming Chiao Tung University, Hsinchu, Taiwan

Abstract. The study, conducted at two regional hospitals in Taichung, Taiwan, aimed to analyze emergency room patient data using Gaussian Mixture Model (GMM) for clustering based on intubation status. Out of 137,722 cases spanning January 1, 2017, to September 30, 2023, 1.14% underwent intubation. The study included the following variables: continuous variables such as WBC (White Blood Cell count), Hb (Hemoglobin), Hct (Hematocrit), MCV (Mean Corpuscular Volume), Blood Sugar, Creatinine levels, HR (Heart Rate), RR (Respiratory Rate), BT (Body Temperature), and SI (shock index). Additionally, categorical variables encompass Gender and Diabetes Mellitus (DM). Patients were divided into Rule In and Rule Out groups, with distinct intubation rates, 2.56% and 0.75%. Rule Out group, with a low intubation rate, identified patients with minimal intubation probability. We can infer that patients with elevated WBC, low Hb, low Hct, high blood sugar, high creatinine, high heart rate, and high shock index are more likely to require intubation compared to patients with normal values. Further research is needed to explore its application.

Keywords: Gaussian Mixture Model · clustering analysis · emergency room · intubation

1 Introduction

Intubation prediction is crucial in rapidly escalating conditions. Several studies have explored machine learning (ML) and statistical models to forecast the need for intubation, aiding in timely intervention and resource allocation. Bolourani et al. [1] developed an XGBoost model achieving 0.77 AUC in predicting respiratory failure within 48 h of admission, emphasizing predictors such as oxygen delivery method, age, ESI, respiratory rate, lactate levels, and demographics. Venturini et al. [2] introduced cure-ML for early intubation prediction in ICUs, while Stefan et al. [3] developed clinical risk scores for NIV failure prediction, and Gaudet et al. [4] derived the CERES score for late respiratory

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 3–10, 2024.
https://doi.org/10.1007/978-3-031-66538-7_1

failure prediction in severe COVID-19 cases. Additionally, Arvind et al. [5] developed an ML algorithm for identifying high-risk COVID-19 patients, and Sakai et al. [6] identified markers for impending mechanical ventilation in emergency room COVID-19 patients, underscoring the importance of blood glucose and SpO2/FiO2 ratio in risk assessment. However, there is no research using clustering methods to classify emergency department patients based on whether they will be intubated. This study attempts to use GMM to cluster emergency department patients and explore the differences between clusters.

2 Method

2.1 Subject Selection

In a retrospective analysis conducted at two Taichung, Taiwan regional hospitals, patient data from emergency departments were collected from January 1, 2017, to September 30, 2023, totaling 246,259 cases. Exclusions comprised 22,981 pediatric cases, 53,521 cases lacking blood pressure measurements or with recorded values of zero, and 32,035 cases without blood test reports, resulting in a final sample size of 137,722 cases. Among these, 1,570 cases (1.14%) intubated [7].

2.2 Clustering Algorithms and Clustering Variables

Using a Gaussian Mixture Model, we clustered all emergency department patients to explore potential differences in intubation rates across sub-groups. The clustering variables comprised continuous measures such as WBC, Hb, Hct, MCV, Blood Sugar, Creatinine levels, Heart Rate, RR, BT, and shock index, alongside categorical variables like Gender (0 for female, 1 for male) and Diabetes Mellitus (DM, 1 for presence, 0 for absence).

2.3 Using AIC and BIC to Determine the Optimal Number of Clusters

Firstly, we standardized the data intended for clustering. Subsequently, we applied the Gaussian Mixture Model to partition the data into different numbers of clusters ranging from 1 to 11. The AIC and BIC values were computed for each cluster configuration. These values were then plotted as a line graph to observe the insensitivity of the data after being segmented into K clusters. Specifically, even as the value of K increased, the rate of decrease in AIC and BIC became minimal, resulting in the slope of the line approaching stability. This is visually represented on the line graph as an approximately horizontal state [8] (Fig. 1).

The Akaike Information Criterion (AIC), devised by Hirotugu Akaike, serves as a statistical tool for model selection and comparison by balancing a model's goodness of fit with its complexity.

The AIC is computed as follows:

$$AIC = 2k - 2\ln(L) \tag{1}$$

where L : likelihood of the model given the data, k : number of parameters in the model.

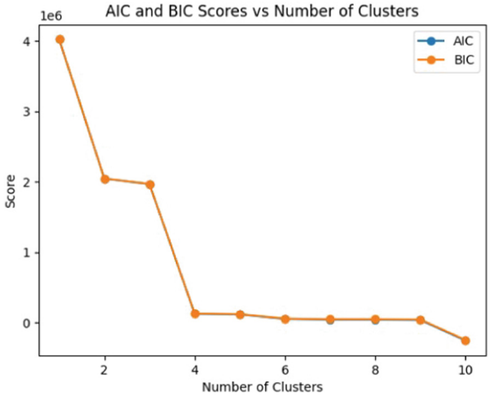


Fig. 1. AIC and BIC Scores vs Number of Clusters

The Bayesian Information Criterion (BIC) is a statistical measure used in model selection and evaluation, particularly in the context of Bayesian statistical inference. The formula for BIC is given by:

$$BIC = -2\ln(L) + k \cdot \ln(n) \tag{2}$$

where L : likelihood of the model given the data, k : number of parameters in the model, n : sample size.

2.4 The Selection of Cluster Numbers

After examining AIC and BIC scores plotted against cluster numbers, we found stable patterns when segmenting the data into 2 or 4 clusters, indicating meaningful segmentation. Dividing patients into two groups revealed one with a higher intubation rate (2.56%) and another with a lower rate (0.75%). Further division into four groups showed clinical insignificance in the highest intubation rate group, while merging lower and higher rate groups mirrored the distribution of the two-group division. Hence, we opted for the simpler two-group division for clearer clinical interpretation, identifying a 'Rule In' group with higher intubation probability and a 'Rule Out' group with very low intubation probability.

2.5 Using Two Sample t Test to Assess Whether There Are Significant Differences Between Expected Outcomes and Variables After Clustering

Firstly, we use two sample t test to compare continuous variables, where:

$$\mathrm{t} = \frac{|\bar{\mathrm{X}}\mathrm{i} - \bar{\mathrm{X}}\mathrm{p}|}{\sqrt{\frac{Si^2}{Ni} + \frac{Sp^2}{Np}}} \ or \ \ \mathrm{t} = \frac{|\bar{\mathrm{X}}\mathrm{o} - \bar{\mathrm{X}}\mathrm{n}|}{\sqrt{\frac{So^2}{No} + \frac{Sn^2}{Nn}}}$$

where $\bar{X}_i$, $\bar{X}_p$, $\bar{X}_o$, $\bar{X}_n$: Mean values of Rule In group, Positive group, Rule Out group, and Negative group, respectively. S_i, S_p, S_o, S_n: Standard deviations of Rule In group, Positive group, Rule Out group, and Negative group, respectively. N_i , N_p, N_o, N_n: Number of observations in the Rule In group, Positive group, Rule Out group, and Negative group after grouping.

In addition, we use two sample t test to compare categorical variables, where:

$$\mathrm{t}\ =\frac{|\mathrm{Ni}\ -\ \mathrm{Np}|}{\sqrt{\frac{(\mathrm{Ni}+\mathrm{Np})(2N-\mathrm{Ni}\ -\ \mathrm{Np})}{N}}}\ or\ \ \mathrm{t}\ =\frac{|\mathrm{No}\ -\ \mathrm{Nn}|}{\sqrt{\frac{(\mathrm{No}+\mathrm{Nn})(2N-\mathrm{No}\ -\ \mathrm{Nn})}{N}}}$$

where N: Total cases number, *Ni , Np, No, Nn*: Number of observations in the Rule In group, Positive group, Rule Out group, and Negative group after grouping.

$$\mathrm{P}-\mathrm{value}=2*\left[1-\mathrm{CDF}(t,df)\right] \tag{3}$$

where CDF(*t, df*) : The Cumulative Distribution Function computes the probability of a random variable, like the t-distribution with a specified degrees of freedom (*df*), being less than or equal to a specific value (*t*). P-value : A two-tailed p-value calculated from CDF(*t, df*).

3 Result

3.1 Using GMM for Clustering to Obtain Data for the Four Groups: ETI, not ETI, Rule In, and Rule Out

Patients who have been intubated are classified as the ETI group, while patients who have not been intubated are classified as the Not ETI group. After GMM clustering into two groups, the Rule In group, with a higher intubation rate, comprised 29,679 cases (2.56% intubated in the Rule In group), and the Rule Out group, with a lower intubation rate, comprised 108,043 cases (only 0.75% intubated in the Rule Out group).

3.2 Comparison of Various Clinical Variables Between the Intubated and Non-intubated Groups

We conducted analyses to identify differences between intubated and non-intubated groups. Utilizing Two Sample t-tests for continuous variables, only body temperature showed no significant difference, while all other variables exhibited significant disparities. Then, we compared categorical variables and found significant differences in proportions between intubated and non-intubated cases across all groups: Rule Out, Rule In, Male, Female, With DM (Diabetes Mellitus), and Without DM. Chi-square tests confirmed these findings, reaffirming significant distinctions in proportions across the six groups.

3.3 Comparison of Various Clinical Variables Between the ‘Rule In’ and ‘Rule Out’ Groups

We conducted comparisons between continuous variables of the ‘Rule In’ and ‘Rule Out’ groups, revealing significant differences across all variables using Two Sample t-tests. Additionally, we examined categorical variables between these groups, finding significant disparities in proportions of intubated and non-intubated cases within most categories, except for Male and Female groups. Specifically, significant differences were observed in proportions across Intubated, Non-intubated, With DM, and Without DM categories. These results were confirmed through chi-square tests across the six groups, indicating consistent disparities between the 'Rule In' and 'Rule Out' groups.

3.4 Compare the Differences in Confidence Intervals Among the Four Groups of Continuous Variables: Intubated, Non-Intubated, Rule In, and Rule Out

We aimed to illustrate distribution trends of various continuous variables using confidence intervals. Table 1 presents the ranges of Confidence Intervals for each of the four groups: Intubated (ETI), Non-intubated (Not ETI), Rule In (RI), and Rule Out (RO). These intervals were plotted into a chart, as depicted in Fig. 2. While the chart provides a visual indication of distribution trends, particularly the extent of overlap between Confidence Intervals of the RO and Not ETI groups, comparing solely through charts has limitations. We can sense trends but cannot confirm differences' existence or magnitude. To address this, we employed Two Sample t-tests to quantify differences between groups and assess their statistical significance.

Table 1. Confidence Intervals between the groups of Intubated, Non-intubated, Rule In, and Rule Out

	ETI CI	Not ETI CI	Rule_In CI	Rule_Out CI
WBC	(12476.75, 13812.17)	(9907.65, 9965.71)	(11063.54, 11276.9)	(9620.76, 9668.13)
Hb	(12.38, 12.66)	(12.92, 12.94)	(11.53, 11.59)	(13.29, 13.31)
Hct	(37.56, 38.37)	(38.38, 38.46)	(34.55, 34.72)	(39.42, 39.49)
MCV	(90.15, 91.04)	(88.34, 88.43)	(88.97, 89.15)	(88.19, 88.28)
B/S	(197.55, 209.54)	(152.39, 153.34)	(228.4, 231.94)	(132.13, 132.61)
Cr	(1.86, 2.08)	(1.34, 1.36)	(2.83, 2.9)	(0.94, 0.95)
HR	(101.79, 104.56)	(92.55, 92.77)	(97.34, 97.87)	(91.33, 91.58)
RR	(22.78, 23.5)	(19.8, 19.83)	(21.02, 21.13)	(19.51, 19.53)
BT	(36.68, 36.82)	(36.76, 36.77)	(36.79, 36.82)	(36.75, 36.76)
SI	(0.79, 0.82)	(0.68, 0.68)	(0.73, 0.74)	(0.67, 0.67)

3.5 Utilize Two Sample t Test to Quantify the Differences Between the Two Groups in Comparison

In our analysis, we used two-sample t-tests to compare the RI and ETI groups, resulting in t_i, and the RO and Not ETI groups, resulting in t_o. For temperature, t_i had a non-significant p-value (>0.05), indicating no significant difference between RI and ETI. However, significant differences were found for other variables. Sorting t_i and t_o values revealed trends, with smaller values suggesting narrower disparities and larger values indicating greater disparity between groups, consistent with our hypothesis.

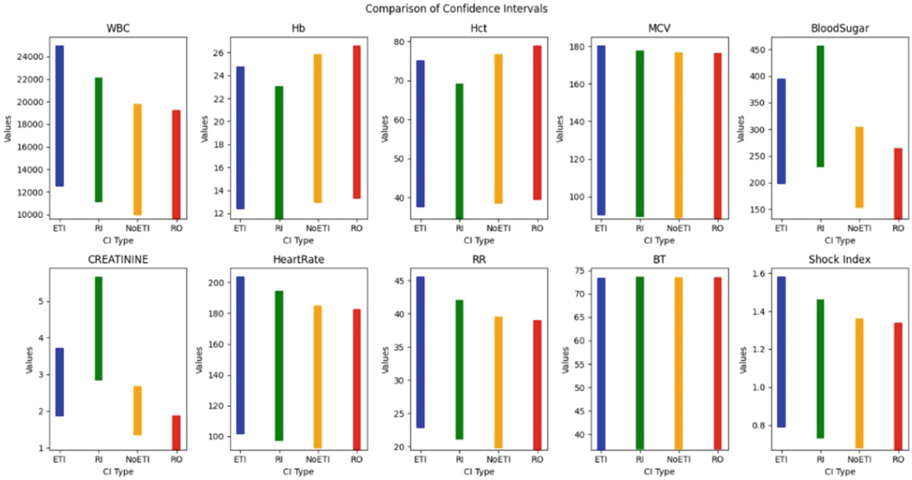


Fig. 2. Comparisons of Confidence Intervals

4 Discussion

4.1 The Rule Out Group Can Assist Clinicians in Identifying Patients with a Very Low Probability of Intubation

The Rule Out group, encompassing 108,043 cases, represents 78.45% of the total study population, yet its intubation rate remains merely at 0.75%. Consequently, patients meeting the Rule Out group's criteria exhibit a notably high probability of not requiring intubation in the emergency department, standing at 99.25%, with only a 0.75% chance of intubation. Essentially, when a patient's metrics align with specified ranges for variables such as WBC, Hb, Hct, MCV, blood sugar, creatinine, BT, heart rate, and shock index, the probability of intubation is exceedingly low.

4.2 The Differences Between the Rule Out Group and the Group of Patients Intubated in the Emergency Department (ETI Group)

When comparing average values between the Rule Out and ETI groups, we notice that the Rule Out group exhibits more normal values for WBC, Hb, Hct, blood sugar, creatinine, heart rate, and shock index, with MCV and BT showing similar values. This observation aligns with clinical experience. Thus, patients with elevated WBC, low Hb, low Hct, high blood sugar, high creatinine, high heart rate, and high shock index are more prone to require intubation compared to those with normal values.

4.3 The Limitations of This Study

This study is a retrospective review of medical records, which may result in limited data availability and a small number of cases due to missing data. Additionally, certain variables could not be included for evaluation. Future directions for improvement could include conducting prospective studies, optimizing data collection and integration methods, increasing the sample size, incorporating additional variables, exploring different clustering methods, and comparing the performance of different clustering methods. Furthermore, the application of the *t* values in the analysis of clustering results could also be explored.

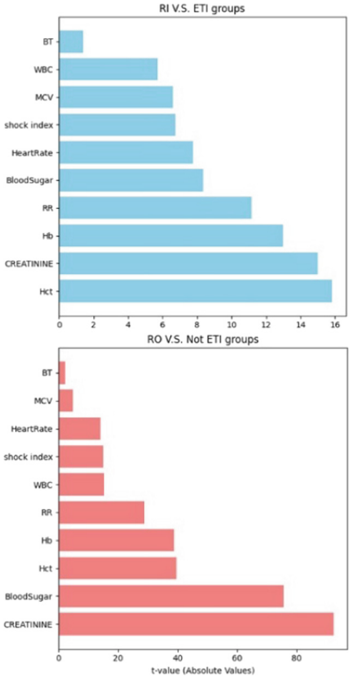


Fig. 3. Sorting the *t* values from smallest to largest

5 Conclusion

This study employed GMM for patient clustering and identified that patients with elevated WBC, low Hb, low Hct, high blood sugar, high creatinine, high heart rate, and high shock index are more prone to require intubation compared to those with normal values. Furthermore, it introduced two-sample t-tests to quantify inter-group differences within the same variable post-clustering. Further investigation is warranted to evaluate the significance and practical implications of these inter-group differences following clustering (Fig. 3).

Disclosure of Interests. To the best of our knowledge, the authors have no conflict of interest, financial or otherwise

Declaration of Generative AI and AI-Assisted Technologies in the Writing Process. During the preparation of this work the authors used ChatGPT 3.5 in order to improve readability. After using ChatGPT 3.5, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

References

1. Bolourani, S., et al.: A machine learning prediction model of respiratory failure within 48 hours of patient admission for COVID-19: model development and validation. J. Med. Internet Res. **23**(2), e24246 (2021)
2. Venturini, M., Van Keilegom, I., De Corte, W., Vens, C.: A novel survival analysis approach to predict the need for intubation in intensive care units. In: Michalowski, M., Abidi, S.S.R., Abidi, S. (eds.) AIME 2022. LNCS, vol. 13263, pp. 358–364. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-09342-5_35
3. Stefan, M.S., et al.: A scoring system derived from electronic health records to identify patients at high risk for noninvasive ventilation failure. BMC Pulm. Med. **21**, 52 (2021)
4. Gaudet, A., et al.: Derivation and validation of a predictive score for respiratory failure worsening leading to secondary intubation in COVID-19: the CERES score. J. Clin. Med. **11**, 2172 (2022)
5. Arvind, V., et al.: Development of a machine learning algorithm to predict intubation among hospitalized patients with COVID-19. J. Crit. Care **62**, 25–30 (2021)
6. Sakai, K., et al.: Combining blood glucose and SpO2/FiO2 ratio facilitates prediction of imminent ventilatory needs in emergency room COVID-19 patients. Sci. Rep. **13**, 22718 (2023)
7. Chang, H., et al.: Machine learning-based suggestion for critical interventions in the management of potentially severe conditioned patients in emergency department triage. Sci. Rep. **12**, 10537 (2022)
8. Boussen, S., et al.: Triage and monitoring of COVID-19 patients in intensive care using unsupervised machine learning. Comput. Biol. Med. **142**, 105192 (2022)

Bayesian Neural Network to Predict Antibiotic Resistance

Laurent Vouriot[1](✉), Stanislas Rebaudet[1,2], Jean Gaudart[1,3], and Raquel Urena[1]

[1] Aix Marseille Univ., IRD, INSERM, SESSTIM, Aix Marseille Institute of Public Health, ISSPAM, Marseille, France
laurent.vouriot@univ-amu.fr
[2] Hôpital Européen, Marseille, France
[3] Sante Publique France, Marseille, France

Abstract. Antimicrobial resistance is recognized by the World Health Organization (WHO) as a significant global health threat. The accurate identification of bacterial susceptibility to antibiotics is crucial, but it often takes several days. On the other hand, in medical decision support systems, such as the one proposed in this contribution, it is crucial to assess the uncertainty of the model when a decision is provided. In this work, we propose a model based on a Bayesian Neural Network to predict antibiotic resistance at different stages of the antibiogram process for a set of 47 antibiotic therapies. Excellent results were achieved, with the area under the receiver operating curve reaching up to 0.9 at the final stage, while also providing a measure of the epistemic uncertainty. To enable clinical usage of the proposed approach as a decision support system, the model has been integrated into a user-friendly and responsive web application accessible on both mobile phones and desktops.

Keyword: Bayesian Neural Networks · Antibiotic Resistance · Deep Learning

1 Introduction

Antimicrobial resistance has been declared by the World Health Organization (WHO) as one of the gravest threats to global health [12]. The best way to control the spread of resistance lies in prescribing the most appropriate treatment. An initial antimicrobial treatment is often prescribed empirically, while awaiting the results of bacterial culture and antibiogram—a process that can take several days, which represents a challenge even for experienced physicians. Here, we propose the first approach using patient metadata and antibiogram to predict bacterial resistance to a wide range of antibiotics and at the successive stages of the antibiogram process [11]. Deep Learning (DL) offers a viable solution, but in a field such as medicine, it is crucial to be cautious and precise. It is difficult to understand its decision process and measure uncertainty. Various machine learning (ML) algorithms have been proposed to address the problem of antibiotic resistance [10, 13]. In this work, we propose the use of Bayesian Neural Networks (BNNs),

Supported by organization SESSTIM.

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 11–16, 2024.
https://doi.org/10.1007/978-3-031-66538-7_2

a probabilistic approach of DL in which standard neural network weights are replaced with probability distributions learned through Bayesian inference [7, 8]. BNNs allow for the quantification of epistemic uncertainty, enabling the model to indicate its level of certainty in its predictions. The utility of uncertainty in ML applied to health has been of real interest lately [1, 5]. The data extracted from antibiograms, accumulated over time, is thus sequential. To align ourselves with real-life applications, the developed model can produce predictions at each stage of the antibiogram process.

2 Methods

2.1 Data

The dataset comprised 91,061 antibiograms collected between 2014 and 2022 in two Marseille hospitals. It includes metadata, bacterial species, and susceptibility to 47 antibiotics. One key challenge was structuring the decision-making process into five stages, akin to clinical practice. Here's an overview of these stages: When the physician encounters a patient suffering from a bacterial infection, he collects patient information (site of infection, patient history of multidrug-resistant bacterial (MDR) carriage, and the medical unit) and send a sample for bacteriological analysis at the laboratory (stage 1). When receiving the sample the bacteriologist makes a direct observation (Stage 2) and then put the sample into culture. Over a period of up to 2 days, the culture grows bacteria, and the results are then sent back to the physician (stage 3). After some more days, the exact specie is identified (stage 5), sometimes only the genus is available (stage 4). The final step is the antibiogram itself, telling for a set of antibiotics if the bacteria is resistant or not, which constitutes our target variables. In alignment with medical practice, the proposed model can provide predictions at each step of the process, aiding in empirical prescription.

2.2 Bayesian Neural Networks

Bayesian Neural Networks can be understood as a combination of neural networks and Bayesian inference. More precisely, they are defined as stochastic artificial neural networks trained using Bayesian inference techniques [4], a statistical inference method based on Bayes' theorem. In traditional DL, weights are fixed values (initially random) that are iteratively updated using gradient descent to minimize a loss function. In contrast, BNNs learn distribution parameters over weights; these distribution parameters are optimized using Bayesian inference methods. This approach presents several notable advantages, such as mitigating overfitting, allowing learning from small datasets, and providing a measure of uncertainty over the predictions [9]. Blundel et al. [2] introduced an algorithm to be able to do backpropagation with bayesian inference (i.e. learn the distributions weights). In practice, we compute the prediction $\hat{y}$ w.r.t an entry x by calculating the mean over a sample of the predictive distribution $\hat{y} = \frac{1}{K}\sum_{k=1}^{K} \mathcal{F}_{w_k}(x)$ with K being the number of samples, and $\mathcal{F}_{w_k}$ being the model with a set w_k of weights sampled from the posterior while also providing a measure of the epistemic uncertainty of the model (i.e., a forward pass). This approach of calculating the empirical mean over a sample of the predictive distribution is analogous to a specific case of ensemble learning. Each forward pass represents the outcome of a unique model with a distinct set of weights. The variance of the predictive distribution is then a measure of the epistemic uncertainty. Each $\mathcal{F}_{w_k}$ in the ensemble corresponds to a different classification decision boundary; therefore, if the model is confident about its predictions, all the decision boundaries will be similar.

The Proposed Model. The used architecture is displayed in Appendix A. After hyperparameters tuning using hyperband [6], best results were obtained with 3 dense variational layers, 300 units each and batch normalization between each layer. In Bayesian neural networks, the prior $p(w)$ acts as a form of regularization. We used a multivariate normal with a diagonal covariance matrix. This prior distribution is analogous to L2 regularization. Regarding the posterior distribution, using an independent normal distribution is a common practice for its mathematical convenience and enabling the usage of the reparameterization trick [4]. The model must output sensibility prediction for a set of 47 antibiotics, thus the last layer of the model is a fully connected layer, with a sigmoid activation to output independent predictions for each antibiotic. For binary classification tasks, the usual loss function used is the binary cross-entropy function. The model was cross-validated on 5 folds of the whole dataset, with each fold being trained for 30 epochs. To assess the model's performance, the Area Under the Receiver Operation Curve (AUROC) was used.

Table 1. AUROC and AUSE on the test set for the BNN and the baseline RNN, for each stage.

	Stage 1	Stage 2	Stage 3	Stage 4	Stage 5	Overall mean
BNN	0.669	0.815	0.861	0.886	0.886	0.8234
RNN	0.676	0.82	0.865	0.896	0.896	0.831
AUSE	0.13	0.066	0.044	0.034	0.034	0.061

2.3 AUROC Scores

Results of the BNN are compared with the best traditional DL model developed, which is a RNN. Both models have identical behaviours across stages, which signifies that the BNN converges almost identically as the RNN. It differs in the scores, which are slightly lower for the BNN. This loss of performance can be expected when doing Bayesian inference. The AUROC for each stage averaged across all antibiotics can be found in Table 1. Generally, the AUROC increases as the stages progress, which is coherent, as more information is gained throughout the antibiogram process. The first stage performs the least effectively (AUROC 0.67) and exhibits the highest variance compared to the other stages, due to the absence of information about the bacteria. At stage 2, there is a significant rise in AUROC (+14%). From this stage onward, the gap between stages narrows, reaching a plateau at 0.896. There are no differences between Stages 4 and 5, where the genus and species features are introduced, the model does not gain additional information from these two features. This is due to the fact that the specie is a more detailed version of the genus, but are nonetheless close features. The Genus was used for its potential utility on rare cases.

2.4 Uncertainty Estimation

To measure the quality of the uncertainty estimations, a common technique is using sparsification plots [3]. The Area Under the Sparsification Error (AUSE) introduced in [3] quantifies the difference between an oracle and the model's sparsification plots by computing the area between the two curves; ideally, the AUSE should be as small as possible. As displayed in Fig. 1, it is generally low. Similarly to the AUROC, it decreases at each stage, indicating that the uncertainty gets more calibrated as we gain information on the bacteria. Thus, the uncertainty measure is meaningful and higher when the model commits mistakes, which is, to some extent, a satisfying behavior.

3 Discussion

The achieved results are generally good, with AUROCs reaching up to 0.9 for the last stages (Table 1). We believe that having a measure of epistemic uncertainty can have a significant impact in practice, as indicated by the well-calibrated uncertainties showcased by the AUSE. However, it is noteworthy that the uncertainty tends to be lower for false positives than for false negatives, with false negative being the mistakes we must avoid the most. It is plausible that some incorrect predictions may come from rare and complex feature combinations inherent in the dataset, suggesting a need for estimating aleatoric uncertainty in conjunction with the predictions. Furthermore, using a broader dataset with additional features could potentially enhance the quality of predictions while reducing overall uncertainty. Despite several attempts with different traditional neural networks, achieving similar scores indicates that the dataset may have reached its maximum predictive capacity. Ideally, we would aim for an even lower AUSE, particularly for the early stages. However, given the limited information available in these stages, perfect calibration may be challenging to attain. Nevertheless, the model remains usable in practice, demonstrating its utility despite these challenges. Even if there is a slight loss of performance compared to the RNN, We consider that it is sufficiently minimal and that the uncertainty represents a significant gain, in favor of the trade off.

4 Conclusion

We present a Bayesian Neural Network (BNN) for predicting antibiotic resistance, leveraging patient metadata data and antibiograms. Our model can generate resistance predictions across all stages of a typical antibiogram process. The Bayesian nature of the model allows for uncertainty quantification, which, upon careful assessment, provides well-calibrated uncertainty measures. We believe that this uncertainty measure constitutes a significant added value, particularly considering the model's intended use in medical practice. Additionally, the model has been integrated into a user-friendly web application, allowing for testing and assessment by professionals.

A Model Diagram

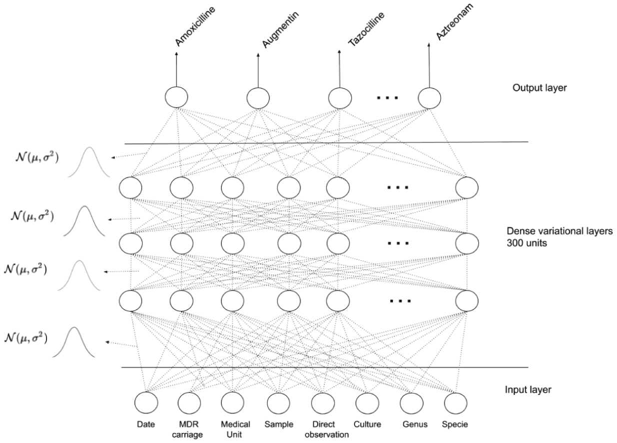


Fig. 1. The architecture of the model is composed of three dense layers, optimized using variational inference, with 300 units each, relu activated. Batch normalization was added between each dense layer

Acknowledgments. This work was supported by a grant co-funded by the Aix-Marseille Institute of Public Health Sciences (ISSPAM) and French National Research Institute for Sustainable Development (IRD). Authors thank the lab bacteriologists of Hôpital Européen and Hopital Saint Joseph who performed the antimicrobial susceptibility testing and extracted the antibiogram dataset.

References

1. Begoli, E., Bhattacharya, T., Kusnezov, D.: The need for uncertainty quantification in machine-assisted medical decision making. Nat. Mach. Intell. **1**(1), 20–23 (2019). https://doi.org/10.1038/s42256-018-0004-1
2. Blundell, C., Cornebise, J., Kavukcuoglu, K., Wierstra, D.: Weight uncertainty in neural networks (2015)
3. Ilg, E., et al.: Uncertainty estimates and multi-hypotheses networks for optical flow (2018)
4. Jospin, L.V., Laga, H., Boussaid, F., Buntine, W., Bennamoun, M.: Hands-on Bayesian neural networks—a tutorial for deep learning users. IEEE Comput. Intell. Mag. **17**(2), 29–48 (2022). https://doi.org/10.1109/mci.2022.3155327
5. Kompa, B., Snoek, J., Beam, A.L.: Second opinion needed: communicating uncertainty in medical machine learning. npj Digit. Med. **4**(1) (2021). https://doi.org/10.1038/s41746-020-00367-3
6. Li, L., Jamieson, K., DeSalvo, G., Rostamizadeh, A., Talwalkar, A.: Hyperband: a novel bandit-based approach to hyperparameter optimization (2018)
7. MacKay, D.J.: A practical Bayesian framework for backpropagation networks. Neural Comput. **4**(3), 448–472 (1992)

8. Neal, R.M.: Bayesian Learning for Neural Networks, vol. 118. Springer, New York (2012). https://doi.org/10.1007/978-1-4612-0745-0
9. Qinghui Yu, J., Creager, E., Duvenaud, D., Bettencourt, J.: Bayesian neural networks. https://www.cs.toronto.edu/~duvenaud/distill_bayes_net/public/
10. Ren, Y., et al.: Prediction of antimicrobial resistance based on whole-genome sequencing and machine learning. Bioinformatics **38**(2), 325–334 (2021). https://doi.org/10.1093/bioinformatics/btab681
11. Urena, R., et al.: Predicting antimicrobial resistance using historical bacterial resistance data with machine learning algorithms (2023). https://doi.org/10.21203/rs.3.rs-2519978/v1
12. World Health Organization: Antibiotic resistance, July 2020. https://www.who.int/news-room/fact-sheets/detail/antibiotic-resistance. Accessed 30 May 2023
13. Yelin, I., et al.: Personal clinical history predicts antibiotic resistance of urinary tract infections. Nat. Med. **25**(7), 1143–1152 (2019). https://doi.org/10.1038/s41591-019-0503-6

Boosting Multitask Decomposition: Directness, Sequentiality, Subsampling, Cross-Gradients

András Millinghoffer[1], Mátyás Antal[1], Márk Marosi[1], András Formanek[1,2], András Antos[1], and Péter Antal[1](✉)

[1] Department of Measurement and Information Systems, Budapest University of Technology and Economics, Magyar Tudósok Körútja 2., 1117 Budapest, Hungary
marosi@mit.bme.hu

[2] Dynamical Systems, Signal Processing and Data Analytics (STADIUS), K.U.Leuven, Kasteelpark Arenberg 10, 3001 Leuven, Belgium

Abstract. The exploration of transfer effects and selection of useful auxiliary tasks in multitask learning and foundation models with downstream tasks remain a largely empirical and computationally demanding process. To reduce the computational cost while maintaining statistical rigor, we investigate (1) the concept of direct transfer effect between tasks, (2) the use of sequential learning to minimize the number of test-train data splits, (3) the possibility of using partial data, and (4) the applicability of gradient-based cross-training task affinities in auxiliary task selection. We apply the methods to a drug-target interaction prediction problem.

1 Introduction

Multitask learning and transfer effects have fundamental importance in artificial intelligence [6], also present in foundational models, general pre-trained transformers, multimodal fusion, and large language models. However, negative transfer effects between tasks are still unresolved [22]. Many recent works concentrated on efficient task decomposition, using various heuristics to decrease the computational cost, such as using partial training data, lightly trained networks, pairwise approximations, and gradient-based task affinities [8,18].

In this paper, we adopt the auxiliary task subset selection approach using hard parameter sharing networks and investigate (1) the concept of multitask Markov boundary to focus the filtering of auxiliary tasks, (2) sequential learning methods to minimize training cost in the auxiliary task search, (3) learning curves to select minimally sufficient sample sizes for auxiliary task screening, and (4) gradient-based cross-training task affinities to speed-up the search process. We demonstrate our methods in a drug-target interaction (DTI) prediction problem [20] and answer the following questions:

Q1 *MTL transfer maps*: What is the pairwise task landscape, when estimating the 'indirect' or 'direct' transfer effects between the tasks?

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 17–35, 2024.
https://doi.org/10.1007/978-3-031-66538-7_3

Q2 *Multi-armed bandit efficiency*: What is the advantage of using a multi-armed bandit scheme over test-train splits in the search process?
Q3 *Efficient performance estimates*: What is the advantage of active learning methods over test-train splits to provide uniformly good estimates?
Q4 *Subsampling*: For which tasks could we use only partial (e.g., 10–20%) data sets to speed up the training in the search process for good auxiliary sets?
Q5 *Gradient-based task affinity* Could we use gradient-based cross-training task affinities to speed up the selection of auxiliary tasks?
Q6 *Pairwise heuristics* What is the efficiency of greedy pairwise heuristics to design higher order MTL?

2 Related Works

Transfer effects in MTL can be related to shared data, representation, and optimization. We use the hard parameter sharing multiple output multilayer perceptron model in our investigations, which is still the most widely used MTL model in drug-target interaction prediction and it can be viewed as a prototype for general pre-trained models fine-tuned with downstream tasks [6,13,14]. Theoretical analysis of transfer effects between tasks corresponding to 'head' networks using shared representations through a common trunk is an active area of research (see e.g. [10]. Empirical approaches to characterize the inductive aspects of transfer adopted the concepts of learning curves and effective data sizes [12,19]. To avoid negative transfer, various screening and task decomposition methods were proposed [8,18]. The selection problem of auxiliary tasks was also formulated in the multi-armed bandit (MAB) framework, as combinatorial multi-armed bandits [7] and multi-bandits [9]. In the life sciences, the drug-target interaction prediction problem has long been viewed as the ideal case for MTL, however, results are still mixed and negative transfer is persistent [13,21]. Our application domain consists of the nuclear receptors present in the NURA data set, which were already investigated by MTL methods [20].

3 Formalization of the Framework and the Questions

We follow the notation from [10,18,19]. We assume that the MTL algorithm $\mathcal{A}$ selects a multi-task binary decision function $f : \mathbb{R}^d \to \{0,1\}^K$ from a function class $\mathcal{F}$, where d denotes the input dimension and K denotes the number of tasks. The input and output spaces are denoted by $\mathcal{X}$ and $\mathcal{Y}$, and their joint probability distribution is denoted by P_{XY}. Following the hard parameter sharing approach, the function f is defined as a composite mapping

$$f = h \circ g, \text{ with } g \in \mathcal{G} \subset \{g' : \mathbb{R}^d \to \mathbb{R}^r\} \text{ and } h_i \in \mathcal{H} \subset \{h' : \mathbb{R}^r \to \{0,1\}\}. \quad (1)$$

The shared feature map g is frequently called trunk, which provides a common representation for all the classifier functions h_i in a domain, and $h = (h_1, \ldots, h_K)$ denotes the vector of classifiers for each task $i = 1, \ldots K$.

We assume a loss function $L(\hat{y}, y)$ specifying the error for the prediction $\hat{y} = f(x)$ and true value y, where $\hat{y}, y \in \{0,1\}^K$. This allows the definition of an idealized performance measure for a given function f, usually with a loss function decomposable for the tasks

$$L(f) = \mathbb{E}_{P_{XY}}\left[L(f(X), Y)\right] \text{ where } L(f(X), Y) = \sum_{i=1}^{K} L_i(f_i(X), Y_i). \qquad (2)$$

In practice, the learning algorithm $\mathcal{A}$ uses a data set D_n to select a decision function $\hat{f} = \mathcal{A}(D_n)$. The expectation over the data set D_n defines the idealized performance measure of $\mathcal{A}$ with sample size n:

$$L_n^{\mathcal{A}} = \mathbb{E}_{P_{D_n}}\left[L(\mathcal{A}(D_n))\right]. \qquad (3)$$

In practice, $L_n^{\mathcal{A}}(\mathcal{F})$ is estimated using M train-test data sets $[(D_n^{(m)}, {D'}_{n'}^{(m)})]_{m=1}^{M}$

$$\hat{L}_{n,n'}^{\mathcal{A}} = \frac{1}{M}\sum_{m=1}^{M} \hat{L}(\mathcal{A}(D_n^{(m)}), ({D'}_{n'}^{(m)})). \qquad (4)$$

We expect complete x inputs, but in the case of high-dimensional outcomes, outcomes are frequently incomplete. In this case, $\hat{L}(f, D'_n)$ denotes the sum of the respective estimated losses based on the n'_i number of outcomes for Y_i present in the test data set D'_n. For losses, such as the Area Under the Receiver Operating Curve (AUROC) and the Area Under the Precision-Recall Curve (AUPR), the performance can be estimated analogously.

We define the model's performance/loss for task i, when trained with the auxiliary task set s_i, as L_{i,s_i}. There are four main training scenarios: single task learning (STL) denoted $L_{i,-}$ (or L_i), $s_i = \emptyset$; multitask learning (MTL) denoted $L_{i,\sum}$, $s_i = \{1, \dots K\} \setminus \{i\}$; pairwise learning (PW) denoted $L_{i,j+}$, $s_i = \{j\}$ and leave-one-out learning denoted $L_{i,j-}$, $s_i = \{1, ..., K\} \setminus \{i, j\}$.

In this predictive approach, the MTL problem is formalized as follows: for each task i find auxiliary task set s_i minimizing L_{i,s_i}, optionally satisfying that the sum of the corresponding c_{i,s_i} costs kept below a specific b budget (for a related formalization, see [18]). In the paper, we assume that the auxiliary s_i task subsets for a task i are all the $K-1$ singular sets, all the $K-1$ leave-one-out sets with size $K-2$, and the set of all $K-1$ other tasks. *Our primary goal is to identify the best models from these for each task and to estimate the corresponding losses uniformly well.*

3.1 Conditional Markov Boundaries for Direct Transfer Effects

In a probabilistic formalization of the MTL problem, we can exploit that tasks can be conditionally dependent given the input X, if they share internal mechanisms and thus they may be confounded or their missing-not-at-random properties are coupled. This suggests a conditional extension of the well-established concept of Markov blankets for the MTL problem.

Definition 1. *The task set s_i is a conditional Markov blanket of task Y_i, iff $I(Y_i, Y \setminus s_i | X, s_i)$. If s_i is minimal, then it is called a conditional multitask Markov boundary of Y_i ($\mathcal{C}(Y_i)$).*

Note that the leave-one-out sets corresponding to $L_{i,j-}$ allow a test for the membership of task j in the $\mathcal{C}_i$ of task i ($j \in \mathcal{C}_i$), which can be used in a complimentary pairwise heuristics to the MTL problem (see question Q1).

3.2 Multi-armed Bandit and Active Learning-Based Speed-Ups

We focus on the computationally most costly estimation in Eq. 4 and try to minimize the number of evaluations/splits both in selecting the best model and in their loss estimation (see questions Q2 and Q3). Below we illustrate the motivation of using data-dependent exploring strategies in these MAB settings in general. In both settings, we have $K' \geq 2$ arms (depending on the number K of tasks) and arm k is parameterized by a fixed (unknown) distribution ν_k over $[0, 1]$ with mean μ_k. When pulled, its associated reward/sample is drawn at random from ν_k independently of all previous rewards. In our case, each μ_k corresponds to L_{i,s_i} for some s_i, and ν_k corresponds to the distribution of the empirical estimates of this L_{i,s_i} taking the form of Eq. 4.

Let $\mu^* = \mu_{k^*} = \max_{k=1,\ldots,K'} \mu_k$ the mean value of a best arm k^*.

Pure Exploration. Finding a subset s_i that is almost as good as the best allowed subset can be formulated as a pure exploration problem, [3,4]. Here, at round t, the forecaster chooses an arm I_t to be pulled based on past (pulls, rewards) till round $t-1$ (*exploration*), and sees the associated reward Y_t. Then it outputs a *recommendation* J_t based on past till round t. In I_t and J_t external randomization may be used. We evaluate the forecaster through its simple regret at round n, that is, the regret $r_n = \mu^* - \mu_{J_n}$ for arm J_n. For simplicity, assume that there is a unique optimal arm. Introduce the gaps and minimal gap

$$\Delta_k = \mu^* - \mu_k \text{ for } k \neq k^*, \quad \Delta = \Delta_{k^*} = \min_{k \neq k^*} \Delta_k, \quad \text{and } H_1 = \sum_{k=1}^{K'} \frac{1}{\Delta_k^2}. \tag{5}$$

Now, if the forecaster just naively allocated, e.g., uniformly and recommended the empirically best arm (EBA, see also [4]), then for some $\{\nu_k\}$'s, the sample complexity for the simple regret rate would be of order at least $K' H_1 \log(\Delta/\delta)$:

Proposition 1. *For uniform allocation and EBA recommendation for $\Delta \leq 1/\sqrt{4K'}$, there are Bernoulli $\{\nu_k\}$'s such that for all $n \geq K'/\Delta$,*

$$\mathbb{E}\left[r_n\right] > \sqrt{\frac{3K'}{n\pi}} \frac{1}{8 + 3/(4\Delta)} \exp\left(-\frac{16n}{K' H_1}\right).$$

The proof, given in Appendix F along with a detailed version, is based on the well-known fact that one needs of the order of $1/\Delta^2$ samples to differentiate the means of two distributions with gap Δ, as mentioned in [3].

On the other hand, for the Successive Rejects (SR) algorithm described in [3, Fig. 3], we have as an immediate consequence of [3, Theorem 2]

Proposition 2. *With* $\overline{\log}K' = \frac{1}{2} + \sum_{i=2}^{K'} \frac{1}{i}$ *for the expected simple regret of SR*

$$\mathbb{E}\left[r_n\right] \leq \frac{K'(K'-1)}{2} \exp\left(-\frac{n-K'}{H_1 \overline{\log} K'}\right).$$

This yields a sample complexity of order at most $K' + H_1 \log K' \log(K'/\delta)$.

Active Learning. Providing uniformly accurate estimates for the performance of all the allowed auxiliary subsets can be formulated as an active learning problem, [1,5]. For details see Appendix E,

3.3 Multi-task Learning Curve Based Speed-Up

The use of a partial data set is a practical choice to accelerate the MTL process, see e.g. [18]. However, as formulated in question Q4, in many real-world problems tasks are not saturated and frequently are in the low data regime. To investigate the applicability of partial data sets in at least learning to rank the candidate auxiliary subsets or to estimate the sign of their effect on the target we used the concept of learning curves (LC). The expected performance at a given sample size in Eq. 3 provides a formal definition for the LCs [19]. We use the following power law approximation:

$$\mathrm{POW3}(n) = c - an^{-b}, \tag{6}$$

where n denotes the sample size, the parameters a and b influence the learning rate, whereas parameter c represents the asymptotic performance limit.

3.4 Gradient-Based Speed-Up

The estimation of inductive transfer is a costly computation; however, transfer effects are also manifested during training. The Task Affinity Groupings (TAG) method was designed to utilize this latter aspect to determine the optimal clustering of tasks for training in multi-task neural networks [8]. Within this framework, all tasks are integrated and trained within a comprehensive model. A key feature of TAG is its assessment of the influence of parameter updates for a specific task on the losses associated with other tasks, a process termed inter-task affinity. Drawing inspiration from meta-learning, TAG employs an approach by adjusting parameters for individual tasks and subsequently examining their effects on other tasks. By gathering data on task interrelations, TAG distinguishes between tasks that exhibit positive and negative interactions. This facilitates the formation of task groupings that amplify inter-task affinity, which in turn augments model performance. We expanded TAG to collect information about all three STL, MTL, PW scenarios used in the paper.

4 Materials and Methods

Following [20], we use the NURA-2021 data set binarized as 'strong binder' versus other labels. It includes 22 targets and 31006 compounds. Since completely random train/test splits suffer from the compound series bias, leading to overoptimistic performance estimations [14], we utilize a more realistic, scaffold-based train/test split in the spirit of [16], which resulted in 6441 scaffolds.

Our definition of the shared parameter model in Eq. 1 follows the SparseChem multiple output MLP model with the decomposition at the penultimate layer as the g foundation model and a ReLU activation in the h for the task heads [2]. Trainings were performed using the SparseChem model, which consisted of a 32000 neuron-wide input layer, a 500 neuron-wide hidden layer, and an output layer the width of which was determined by the number of tasks present in the corresponding learning. ReLU and sigmoid activation functions were used in the first two layers and on the output. Hyperparameters for the training of neural networks were determined by grid search. During their training, ADAM optimization scheme was used with a learning rate of 0.0001. Training was done for a fixed number of 35 epochs.

5 Results

In the following, we present results for the questions Q1-Q6. To assess the performance of the trained models, the following operations were carried out: (1) the set of available scaffolds was split randomly into training and validation folds with sizes 80% and 20%, respectively; (2) the evaluated neural network models were trained for 35 epochs and then evaluated by calculating their AUROC and AUPR measures for each target contained by the corresponding data set. This cycle of fold splitting-training-evaluation was repeated 100 times in order to achieve a confident estimation of the mean and variance of AUROC/AUPR performance. For each task, we evaluated the STL, MTL, and the two types of pairwise scenarios, which means $2+2*(K-1)$ options in the MAB. Throughout the paper, we use the best option for each task as the reference.

5.1 Effect of Multi-task Learning (Question Q1)

We performed a systematic screening over all the tasks for the effect of multi-task learning comparing the performance of the selected task composition settings against single-task performance as a baseline: (1) all the tasks together, (2) two targets together to assess their pairwise interaction, and (3) all the tasks except one in a *leave-one-out* scheme (see Fig. 1).

Overall, multi-task learning improved the AUROC performance for 14 tasks at a nominally significant level (using Welch paired t-test for the assessment) and the AUPR performance also for 14 tasks (with an overlap of 13 tasks).

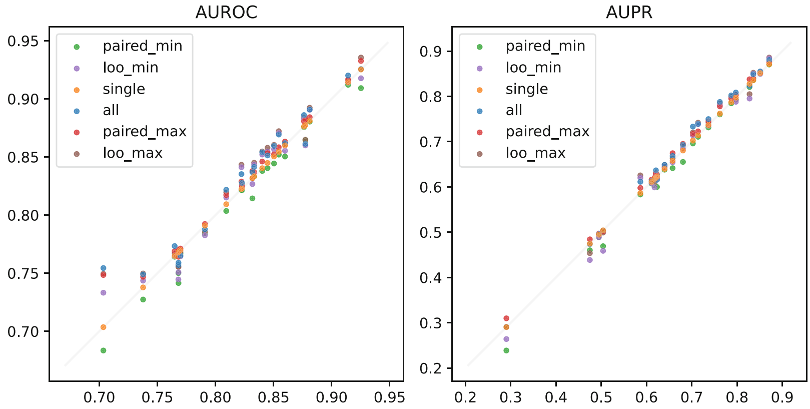


Fig. 1. Task-by-task comparison of AUROC and AUPR performances of the single-task (horizontal axis), multi-task, and the best and worst of the pairwise and leave-one-out cases (vertical axis).

5.2 Effect of Bandits (Question Q2)

We evaluated the UCB algorithm for each task with the STL versus MTL scenarios. Each MAB performed 5, 10,25, 50, and 100 draws and we calculated their accuracy compared to the reference using 100 repeats (see Fig. 3).

5.3 Efficiency of Active Learning (Question Q3)

The efficiency of the active learning algorithms depends on the variances of the arms (see Appendix E), i.e., on the variances of the AUROC and AUPR values for the STL, MTL, and pairwise scenarios. Using the same systematic evaluation with 100 train-test splits, we estimated the variances (Fig. 4 shows the variances (min/max)). For illustrations of the convergence of standard errors, see B.

5.4 Learning Curves of Multitask Learning (Question Q4)

Figure 6 illustrates the LC with parameters for task 12 and shows the relations per task between the reference transfer effects and the transfer effects estimated using only the 10% of the available data set.

To test the applicability of partial data sets in the screening process, we investigated the usability of x% of the data in a simple decision about the STL versus the MTL options and compared this decision to a reference decision based on the 100 train-test splits (see Table 1).

5.5 Gradient-Based Cross-Training Task Affinities (Question Q5)

Analogously to the testing of the learning curves, we investigated the usability of the gradient-based cross-training task affinities calculated using the first m

epochs. Figure 7 illustrates the relation of the gradient-based task affinities and the real differences. Table 2 shows the usability of gradient-based task affinities to discriminate between the STL and the MTL options, compared to a reference decision based on the 100 train-test splits.

5.6 Multivariate Effect of the Positive Pairwise Tasks (Question Q6)

Finally, we selected for each task the auxiliary pairwise tasks with positive effects both with respect to AUROC and AUPR (see Fig. 2).

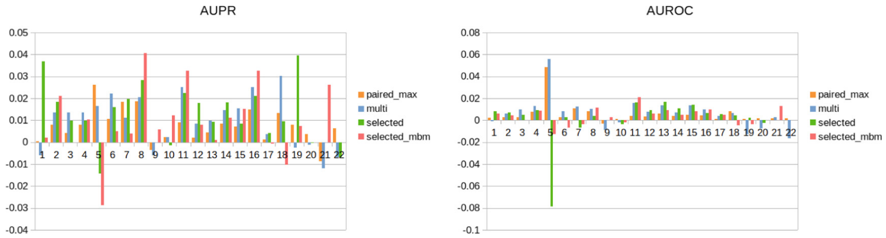


Fig. 2. Effect of the inclusion of the auxiliary tasks with positive effects on AUROC and AUPR (denoted by 'selected') and those auxiliary tasks, the removal of which from the full multitask model has a negative effect (denoted by 'selected_mbm'). The STL scenario is the baseline, for comparison the full multitask learning (multi) and the best pairwise options (paired_max) are also indicated.

6 Discussion

The presented results for questions Q1-Q6 offer systematic answers for the heuristics reported in the multitask decomposition problem. As the results shown in Fig. 1 indicate most of the tasks can be significantly improved even by simple MTL approaches (Q1). Surprisingly, the proposed conditional Markov Boundary-based 'direct' transfer effect in itself proved to be the most advantageous in certain cases, which suggests that negative transfer can be linked to a single detrimental auxiliary task. The tentative applicability of MABs over only the restricted set of scenarios confirmed their efficiency (see Fig. 3), although their full-fledged application should encompass the use of combinatorial bandits in the multi-bandit framework as the auxiliary task sets should be selected for each task simultaneously (Q2). As Fig. 4 illustrates the variances of the performance of the models are widely varying, which is probably related to the sample size differences (Q3). Given the increasing relevance of tasks with limited data and few-shot learning, the significance of this problem is expected to escalate. Consequently, active learning methods may offer substantial utility in addressing these challenges. The relative data scarcity does not allow the use of a partial data set with fixed size in this domain (see Table 1), but learning curves fit at well-selected sample sizes could offer efficient strategies to predict the performance (Q4). Despite its attractively low computational cost, gradient-based cross-training task affinities cannot be used to approximate the reference transfer effects in this problem (Q5). The joint use of auxiliary tasks with positive transfer effects could further improve the performances, indicating the viability of the multitask decomposition approach in the DTI prediction problem (Q6).

7 Conclusion

Transfer effects in multi-task learning present both theoretical and practical challenges. The DTI prediction problem, characterized by its fundamental relevance and industry-scale data sets, exemplifies these challenges. However, the data sets are often highly incomplete, relatively scarce, and missing-not-at-random, with the computational costs of training large multi-task models creating a bottleneck in reliably estimating generalization performance. In response, we explored novel applications of active learning methods aimed at reducing the number of test-train data splits, thereby facilitating more efficient explorations of various heuristics proposed for the multitask decomposition problem. Our findings further suggest that adaptive approaches to test-train data splits enable effective and scalable use of advanced multi-armed bandit methods in this context, searching for optimal auxiliary sets. Efficiently utilized heuristics based on learning curves and gradient-based task affinities from partial training could guide this process, enhancing the overall efficiency of multi-task learning.

Acknowledgments. This study was supported by the Hungarian National Research, Development, and Innovation Office K 139330 grant; the European Union (EU) Joint Program on Neurodegenerative Disease (JPND) Grant SOLID JPND2021-650–233; the National Research, Development, and Innovation Fund of Hungary under Grant TKP2021-EGA-02, the European Union project RRF-2.3.1–21-2022–00004 within the framework of the Artificial Intelligence National Laboratory.

Appendix

The structure of the appendix is as follows:

- Section A provides a comprehensive overview of the background.
- Section B presents results for the sequential learning approaches.
- Section C illustrates data sufficiency using learning curves.
- Section D reports results about the use of gradient-based task affinities.
- Section E gives details and further explanations on the active learning approach
- Section F presents details and proofs about the pure exploration approach.

A Background

In life sciences, the drug-target interaction prediction problem has long been viewed as the ideal case for MTL [30,50,57]. However, results are still mixed and negative transfer is persistent [13,21,38,39,45,58,65]. Transfer effects and transfer learning are intensively investigated in many fields, such as in multi-task learning (MTL) [61], in transfer learning and model transformation/distillation [22,24], in multimodal fusion [25,28,62], in curriculum learning [29], in learning with prior knowledge and few-shot learning [56], in meta-learning [35,37,59], in active learning and adaptive study design [43,51], in sequential learning [54], and in learning downstream tasks with foundation models [12].

The selection of multiple tasks for joint learning motivated a series of sequential learning methods [42]. Combinatorial multi-armed bandits (MABs) allow the pull of multiple arms simultaneously resulting in an aggregate reward of individual arms [7]. Notably, the 'top-k' extension considers the subsets up to size k and allows variants whether the rewards for the individual arms are available or only their non-linear aggregations [23,49]. Another relevant extension to our formalization with multiple auxiliary task subset selection problem for each task is the multi-bandit approach, also allowing overlap between the arms [9,52]. Finally, recent MAB extensions investigated the use of task representations and learning task relatedness [32,46,54]. MABs are successfully applied in hyper-parameter optimization and neural architecture search (see e.g. [34]).

A wide range of techniques was also proposed for MTL-specific optimization [31,40,44,53,63,64], although systematic empirical evaluations indicate very modest or not significant advance [61]. To avoid negative transfer, various screening and task decomposition methods were proposed [8,18,55].

Although the feature subset selection problem offers a direct analogue for the task subset selection problem, this parallel is unexplored. For the introduction and overview of the Markov blanket concept, see [47,48]). For its use in the feature subset selection problem, see e.g. [26,33,41]. For dedicated DTI architectures to explicitly model the DTI mechanisms, see e.g. [27,36,60] Fig. 5.

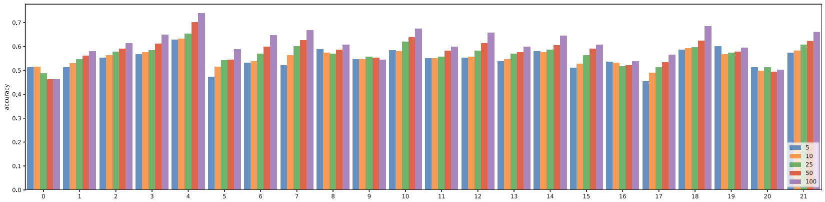


Fig. 3. The accuracy of MABs compared to the reference performing 5, 10, 25, 50, and 100 draws, which is repeated 100 times.

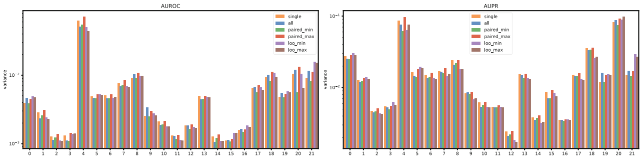


Fig. 4. Histogram of variances of the models per task (only the minimal and maximal variances are shown for the pairwise models).

B Results of Sequential Learning Approaches

C Learning Curves

Table 1. Relations of STL versus MTL decisions estimates using varying amounts of data.

meas	AUROC				AUPR			
training data	$\mathrm{ACC}_{\mathrm{LC}}$	$\mathrm{ACC}_{\mathrm{diff}}$	κ_{LC}	κ_{LC}	$\mathrm{ACC}_{\mathrm{LC}}$	$\mathrm{ACC}_{\mathrm{diff}}$	κ_{LC}	κ_{LC}
10.00%	54.55%	54.55%	0.246	0.246	72.73%	68.18%	0.340	0.364
20.00%	72.73%	77.27%	0.377	0.545	63.64%	81.82%	-0.023	0.560
30.00%	90.91%	86.36%	0.770	0.702	81.82%	86.36%	0.560	0.645
40.00%	86.36%	81.82%	0.592	0.488	81.82%	81.82%	0.488	0.488
50.00%	90.91%	95.45%	0.744	0.879	81.82%	86.36%	0.488	0.582
60.00%	90.91%	90.91%	0.744	0.771	90.91%	90.91%	0.694	0.694
70.00%	95.45%	95.45%	0.879	0.879	90.91%	86.36%	0.694	0.492
80.00%	100.00%	100.00%	1.000	1.000	90.91%	100.00%	0.694	1.000

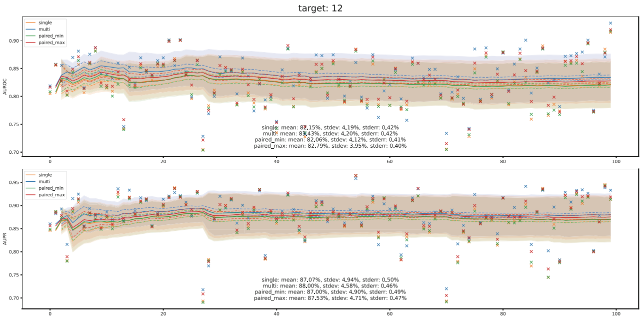


Fig. 5. The evolution of the estimated losses and variances for the models in the case of task 12.

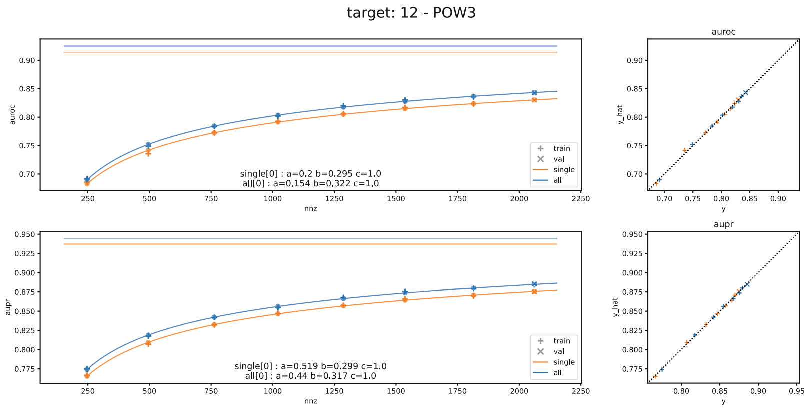


Fig. 6. The STL and MTL learning curves of task 12, the scatter plots on the right depict the fit of the LC function to the actual observations.

D Gradient-Based Cross-Training Task Affinities

Table 2. Accuracy and Cohen's κ values characterizing the goodness of gradient-based affinity scores w.r.t. observed differences in AUROC and AUPR scores, accumulated over 5, 10 and 35 training epochs.

meas	AUROC		AUPR	
epochs	ACC	κ	ACC	κ
5	54.55%	0.241	54.55%	0.241
10	54.55%	0.173	54.55%	0.172
35	59.09%	0.038	68.18%	0.252

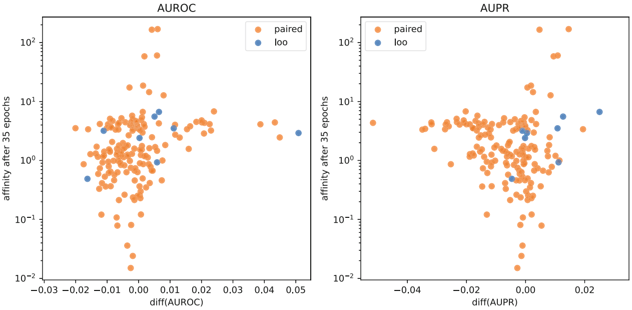


Fig. 7. The relation of affinity scores and actual differences taking into account measurements from the whole (35 epochs) training, for pairwise and leave-one-out settings.

E Active Learning of Performance Measures

Below, we detail the motivation for using data-dependent active learning strategies in general, mentioned at the end of Subsect. 3.2. In this setting, the objective is to estimate all the μ_k's uniformly well given a budget of n pulls. At each round $t \geq 1$, the learner (algorithm) selects an arm k_t to be pulled sequentially, that is, based on past pulls and samples up to round $t-1$ and receives an associated sample Y_t drawn from ν_{k_t} independently of the past. Let $T_{kt} = \sum_{t'=1}^{t} \mathbb{I}\left\{k_{t'} = k\right\}$ be the number of times arm k was pulled up to round t. Denote the sequence of associated samples for arm k also by $X_{k,1}, X_{k,2}, \ldots, X_{k,T_{kn}}$ here, that is, $Y_t = X_{k_t,T_{k_t,t}}$. After round n , the learner returns the empirical estimates

$$\hat{\mu}_{kn} = \frac{1}{T_{kn}} \sum_{t=1}^{T_{kn}} X_{k,t} \qquad (1 \leq k \leq K')$$

of μ_k's (sample means). The accuracy of estimating μ_k by $\hat{\mu}_{kn}$ is measured with its expected squared error (loss) $L_{kn} = \mathbb{E}\left[(\hat{\mu}_{kn} - \mu_k)^2\right]$. The overall performance or loss of the learner to be minimized is measured by the worst loss over the K' arms

$$L_n = \max_{1 \leq k \leq K'} L_{kn}.$$

See [1] for motivation for this loss function.

Consider the non-sequential version of the problem, i.e., the problem of choosing $T_{1n}, \ldots, T_{K'n}$ such that $T_{1n} + \ldots + T_{K'n} = n$ so as to minimize L_n . In this case, due to the independence of samples[1]

$$L_{kn} = \frac{\sigma_k^2}{T_{kn}},$$

where $\sigma_k^2 = \mathrm{Var}\left[X_{k1}\right]$ is the variance of ν_k. So a naive uniform allocation, when each $T_{kn} = n/K'$ (up to rounding), yields

$$L_n = \frac{K' \max_{1 \leq k \leq K'} \sigma_k^2}{n},. \tag{7}$$

loss. Assume for a moment that we know each $\sigma_k^2 > 0$ (for simplicity assumed to be positive). Then there is no value in making the choice of T_{kn}'s data dependent, and the minimizer of L_n is the allocation $\{T_{kn}^*\}_k$ that makes all the losses L_{kn} (approximately) equal, hence (apart from rounding issues)

$$T_{kn}^* = n\frac{\sigma_k^2}{\Sigma^2}.$$

Here $\Sigma^2 = \sum_{k=1}^{K'} \sigma_k^2$, and the value of σ_k^2/Σ^2 gives the *optimal allocation ratio* for arm k. Hence to calculate the optimal allocations, all one needs to know

[1] This equality does not hold when the number of pulls is random, e.g., in adaptive algorithms where the strategy depends on the random observed samples.

about $\{\nu_k\}$ is $\{\sigma_k^2\}$. The corresponding loss is

$$L_n^* = \frac{\Sigma^2}{n},$$

that can be significantly less than that of the uniform allocation in 7 above when σ_k^2's deviates. A good sequential algorithm $\mathcal{A}$ can achieve a loss $L_n(\mathcal{A})$ close to L_n^*, that is, have small *excess-loss* or *regret*

$$R_n(\mathcal{A}) = L_n(\mathcal{A}) - L_n^*,$$

and thus can well overtake the uniform allocation.

For example, [1] provides the GAFS-MAX algorithm and proves that its regret is such that $R_n(\mathcal{A}_{\mathrm{GAFS-MAX}}) = \tilde{O}(n^{-3/2})$.[2] (Note that both the coefficient and the threshold in $\tilde{O}$ depend heavily on the variances through $\Sigma^2 / \min\limits_{1 \le k \le K'} \sigma_k^2$ in the result.)

Another algorithm mentioned in [1], GFSP-MAX, is detailed and empirically shown to have regret of order $\tilde{O}(n^{-3/2})$ in [11].

In [5], two more algorithms were given motivated by the celebrated bandit algorithm UCB, for fixed (known in advance) budget n.

The CH-AS algorithm has again the regret rate $R_n(\mathcal{A}_{\mathrm{CH-AS}}) = \tilde{O}(n^{-3/2})$. (In this result, only the constant coefficient in $\tilde{O}$ depends moderately on the above variance parameter, but not the threshold.)

The other algorithm, B-AS, has also the regret rate $R_n(\mathcal{A}_{\mathrm{B-AS}}) = \tilde{O}(n^{-3/2})$. (Here, the constant coefficient in $\tilde{O}$ has an even weaker dependence on the above variance parameter, but not the threshold. For Gaussian, ν_k's, a regret bound independent of this parameter is also derived.)

Note that an interesting feature that is shared between both the pure exploration and active learning settings is that good strategies should play all the arms as a linear function of n. This is in contrast with the standard stochastic bandit setting, at which the sub-optimal arms should be played logarithmically in n.

F Proof of Proposition 1

First, we restate a more detailed version of the proposition:

Proposition 3. *For uniform allocation, that is, when $I_t = i$ happens n/K'-times (up to rounding) for all i, and EBA recommendation, for any $\Delta < 1/2$, there are Bernoulli $\{\nu_i\}$'s such that for all $n \ge K'/\Delta$,*

$$\mathbb{E}\left[r_n\right] \ge \sqrt{\frac{3K'}{n\pi}} \frac{1}{8 + 3/(4\Delta)} \exp\left(-\frac{16n\Delta^2}{3K'}\right).$$

[2] A nonnegative sequence $\{a_n\}$ is said to be $\tilde{O}(f_n)$, where $\{f_n\}$ is a positive valued sequence, if $a_n \le C\, f_n \log^p(f_n)$ with suitable constants C,$p > 0$.

Whenever $\Delta \le 1/\sqrt{4K'}$ *then also*

$$\mathbb{E}\left[r_n\right] > \sqrt{\frac{3K'}{n\pi}}\frac{1}{8+3/(4\Delta)}\exp\left(-\frac{16n}{K'H_1}\right).$$

Proof. Let ν_i be Bernoulli distributions with parameters

$$\mu_1 = \frac{1+\Delta}{2}, \quad \mu_2 = \frac{1-\Delta}{2}, \quad \mu_3 = \cdots = \mu_{K'} = \frac{\Delta}{2},$$

respectively. Now $k^* = 1$, $\mu^* = \mu_1$, $\Delta_1 = \Delta_2 = \Delta$, $\Delta_3 = \cdots = \Delta_K = 1/2$ by 5. Consequently, $H_1 = 2/\Delta^2 + 4(K'-2)$. We have

$$\mathbb{E}\left[r_n\right] = \mathbb{E}\left[\mu^* - \mu_{J_n}\right] = \sum_{k=2}^{K'} \Delta_k \mathbb{P}\left\{J_n = k\right\} \ge \Delta\mathbb{P}\left\{J_n \ne 1\right\}.$$

For simplicity, we assume that $n = mK'$, so each arm was pulled m times till round n. (Otherwise the proof is similar up to some rounding effect.) Denote here $X_{k,1}$, $X_{k,2}$, $\ldots$, $X_{k,m}$ the sequence of associated rewards for arm k. Recall that EBA recommendation means that

$$J_n \in \operatorname*{argmax}_{i=1,\ldots,K'} \frac{1}{m}\sum_{s=1}^{m} X_{i,s} = \operatorname*{argmax}_{i=1,\ldots,K'} \sum_{s=1}^{m} X_{i,s},$$

so $J_n \ne 1$ follows from $\sum_{s=1}^m X_{1,s} < \sum_{s=1}^m X_{2,s}$, hence

$$\mathbb{E}\left[r_n\right] \ge \Delta\mathbb{P}\left\{\sum_{s=1}^{m} X_{1,s} < \sum_{s=1}^{m} X_{2,s}\right\}.$$

The probability on the right-hand side can be written as

$$\mathbb{P}\left\{\sum_{s=1}^{m}(1-X_{1,s}) + \sum_{s=1}^{m} X_{2,s} > m\right\} = \mathbb{P}\left\{\sum_{s=1}^{m}(1-X_{1,s}) + \sum_{s=1}^{m} X_{2,s} \ge m+1\right\},$$

where we have the sum of $2m$ independent Bernoulli($(1-\Delta)/2$) variables, that is, a Binomial$(2m,(1-\Delta)/2)$ variable. Now an inequality by Slud [17] states that for all $np \le k \le n(1-p)$,

$$\mathbb{P}\left\{\text{Binomial}(n,p) \ge k\right\} \ge \Phi\left(-\frac{k-np}{\sqrt{np(1-p)}}\right),$$

where Φ is the standard normal distribution function. Hence this implies that, whenever $m\Delta \ge 1$, the probability $\mathbb{P}\left\{\text{Binomial}(2m,(1-\Delta)/2) \ge m+1\right\}$ above is at least

$$\begin{aligned}\Phi\left(-\frac{m+1-m(1-\Delta)}{\sqrt{m(1-\Delta)(1-(1-\Delta)/2)}}\right) = \Phi\left(-\frac{1+m\Delta}{\sqrt{m(1-\Delta^2)/2}}\right) &\ge \Phi\left(-\frac{2\sqrt{2m}\Delta}{\sqrt{1-\Delta^2}}\right) \\ &\ge \Phi\left(-4\sqrt{\frac{2m}{3}}\Delta\right)\end{aligned}$$

using $1 \leq m\varDelta$ and $\varDelta \leq 1/2$. A bound in [15] assures that

$$\varPhi(-x) \geq \frac{1}{\sqrt{2\pi}} \frac{x}{x^2+1} e^{-x^2/2},$$

thus

$$\begin{aligned}\varPhi\left(-4\sqrt{\frac{2m}{3}}\varDelta\right) &\geq \frac{4\sqrt{m/3\pi}\varDelta}{\frac{32m\varDelta^2}{3}+1} e^{-16m\varDelta^2/3} \\ &\geq \frac{4\sqrt{3m/\pi}\varDelta}{32m\varDelta^2+3m\varDelta} e^{-16m\varDelta^2/3} = \sqrt{\frac{3}{m\pi}} \frac{4}{32\varDelta+3} e^{-16m\varDelta^2/3}.\end{aligned}$$

Putting together

$$\mathbb{E}\left[r_n\right] \geq \sqrt{\frac{3}{m\pi}} \frac{1}{8+3/(4\varDelta)} e^{-16m\varDelta^2/3}.$$

If $\varDelta \leq 1/\sqrt{4K'}$ then

$$H_1 = 2/\varDelta^2 + 4(K'-2) \leq 2/\varDelta^2 + 1/\varDelta^2 - -8 < 3/\varDelta^2,$$

thus the bound above is further lower bounded by

$$\sqrt{\frac{3}{m\pi}} \frac{1}{8+3/(4\varDelta)} e^{-16m/H_1}.$$

□

References

1. Antos, A., Grover, V., Szepesvári, C.: Active learning in heteroscedastic noise. Theoret. Comput. Sci. **411**(29–30), 2712–2728 (2010)
2. Arany, A., Simm, J., Oldenhof, M., Moreau, Y.: SparseChem: fast and accurate machine learning model for small molecules. arXiv preprint arXiv:2203.04676 (2022)
3. Audibert, J.Y., Bubeck, S., Munos, R.: Best arm identification in multi-armed bandits. In: Proceedings of the Twenty-Third Annual Conference on Learning Theory (COLT'10), pp. 41–53 (2010)
4. Bubeck, S., Munos, R., Stoltz, G.: Pure exploration in finitely-armed and continuous-armed bandits. Theoret. Comput. Sci. **412**(19), 1832–1852 (2011)
5. Carpentier, A., Lazaric, A., Ghavamzadeh, M., Munos, R., Auer, P., Antos, A.: Upper-confidence-bound algorithms for active learning in multi-armed bandits. ArXiv e-prints (Jul 2015), http://arxiv.org/abs/1507.04523
6. Caruana, R.: Multitask learning. Mach. learn. **28**, 41–75 (1997)
7. Cesa-Bianchi, N., Lugosi, G.: Combinatorial bandits. J. Comput. Syst. Sci. **78**(5), 1404–1422 (2012)
8. Fifty, C., Amid, E., Zhao, Z., Yu, T., Anil, R., Finn, C.: Efficiently identifying task groupings for multi-task learning. Adv. Neural. Inf. Process. Syst. **34**, 27503–27516 (2021)

9. Gabillon, V., Ghavamzadeh, M., Lazaric, A., Bubeck, S.: Multi-bandit best arm identification. Adv. Neural Inf. Proc. Syst. **24** (2011)
10. Galanti, T., György, A., Hutter, M.: Improved generalization bounds for transfer learning via neural collapse. In: First Workshop on Pre-training: Perspectives, Pitfalls, and Paths Forward at ICML 2022 (2022)
11. Grover, V.: Active Learning and its Application to Heteroscedastic Problems. Master's thesis, University of Alberta (2009)
12. Hernandez, D., Kaplan, J., Henighan, T., McCandlish, S.: Scaling laws for transfer. arXiv preprint arXiv:2102.01293 (2021)
13. Heyndrickx, W., Arany, A., Simm, J., Pentina, A., Sturm, N., et al.: Conformal efficiency as a metric for comparative model assessment befitting federated learning. Artif. Int. Life Sci. **3**, 100070 (2023)
14. Mayr, A., et al.: Large-scale comparison of machine learning methods for drug target prediction on ChEMBL. Chem. Sci. **9**(24), 5441–5451 (2018)
15. Peric, Z.H., Nikolic, J.R., Petkovic, M.D.: Class of tight bounds on the Q-function with closed-form upper bound on relative error. Math. Methods Appl. Sci. **42**, 1786–1794 (2019)
16. Simm, J., et al.: Splitting chemical structure data sets for federated privacy-preserving machine learning. J. cheminformatics **13**(1), 1–14 (2021)
17. Slud, E.: Distribution inequalities for the binomial law. Ann. Probab. **5**, 404–412 (1977)
18. Standley, T., Zamir, A., Chen, D., Guibas, L., Malik, J., Savarese, S.: Which tasks should be learned together in multi-task learning? In: International Conference on Machine Learning, pp. 9120–9132. PMLR (2020)
19. Viering, T., Loog, M.: The shape of learning curves: a review. IEEE Trans. Pattern Analysis Mach. Intell. **45**(6), 7799–7819 (2022)
20. Wang, J., Lou, C., Liu, G., Li, W., Wu, Z., Tang, Y.: Profiling prediction of nuclear receptor modulators with multi-task deep learning methods: toward the virtual screening. Briefings Bioinform. **23**(5), bbac351 (2022)
21. Xu, Y., Ma, J., Liaw, A., Sheridan, R.P., Svetnik, V.: Demystifying multitask deep neural networks for quantitative structure-activity relationships. J. Chem. Inf. Model. **57**(10), 2490–2504 (2017)
22. Zhang, W., Deng, L., Zhang, L., Wu, D.: A survey on negative transfer. IEEE/CAA J. Automatica Sinica **10**(2), 305–329 (2022)
23. Agarwal, M., Aggarwal, V., Umrawal, A.K., Quinn, C.J.: Stochastic top k-subset bandits with linear space and non-linear feedback with applications to social influence maximization. ACM/IMS Trans. Data Sci. (TDS) **2**(4), 1–39 (2022)
24. Antal, P., Fannes, G., Timmerman, D., Moreau, Y., De Moor, B.: Bayesian applications of belief networks and multilayer perceptrons for ovarian tumor classification with rejection. Artif. Intell. Med. **29**(1–2), 39–60 (2003)
25. Antal, P., Fannes, G., Timmerman, D., Moreau, Y., De Moor, B.: Using literature and data to learn bayesian networks as clinical models of ovarian tumors. Artif. Intell. Med. **30**(3), 257–281 (2004)
26. Antal, P., Millinghoffer, A., Hullám, G., Szalai, C., Falus, A.: A bayesian view of challenges in feature selection: feature aggregation, multiple targets, redundancy and interaction. In: New Challenges for Feature Selection in Data Mining and Knowledge Discovery, pp. 74–89. PMLR (2008)
27. Bai, P., Miljković, F., John, B., Lu, H.: Interpretable bilinear attention network with domain adaptation improves drug-target prediction. Nature Mach. Intell. **5**(2), 126–136 (2023)

28. Baltrušaitis, T., Ahuja, C., Morency, L.P.: Multimodal machine learning: a survey and taxonomy. IEEE Trans. Pattern Anal. Mach. Intell. **41**(2), 423–443 (2018)
29. Bengio, Y., Louradour, J., Collobert, R., Weston, J.: Curriculum learning. In: Proceedings of the 26th Annual International Conference on Machine Learning, pp. 41–48 (2009)
30. Cai, C., et al.: Transfer learning for drug discovery. J. Med. Chem. **63**(16), 8683–8694 (2020)
31. Chen, Z., Badrinarayanan, V., Lee, C.Y., Rabinovich, A.: GradNorm: gradient normalization for adaptive loss balancing in deep multitask networks. In: International Conference on Machine Learning, pp. 794–803. PMLR (2018)
32. Du, Y., Huang, L., Sun, W.: Multi-task representation learning for pure exploration in linear bandits. In: International Conference on Machine Learning, pp. 8511–8564. PMLR (2023)
33. Friedman, N., Koller, D.: Being bayesian about network structure. a bayesian approach to structure discovery in bayesian networks. Mach. Learn. **50**, 95–125 (2003)
34. Guo, H., Pasunuru, R., Bansal, M.: AutoSeM: automatic task selection and mixing in multi-task learning. arXiv preprint arXiv:1904.04153 (2019)
35. Hospedales, T., Antoniou, A., Micaelli, P., Storkey, A.: Meta-learning in neural networks: a survey. IEEE Trans. Pattern Anal. Mach. Intell. **44**(9), 5149–5169 (2021)
36. Huang, K., Xiao, C., Glass, L.M., Sun, J.: MolTrans: molecular interaction transformer for drug-target interaction prediction. Bioinformatics **37**(6), 830–836 (2021)
37. Klein, A., Falkner, S., Springenberg, J.T., Hutter, F.: Learning curve prediction with bayesian neural networks. In: International Conference on Learning Representations (2016)
38. Li, X., et al.: Deep learning enhancing kinome-wide polypharmacology profiling: model construction and experiment validation. J. Med. Chem. **63**(16), 8723–8737 (2019)
39. Lin, S., Shi, C., Chen, J.: GeneralizedDTA: combining pre-training and multi-task learning to predict drug-target binding affinity for unknown drug discovery. BMC Bioinform. **23**(1), 1–17 (2022)
40. Liu, S., Liang, Y., Gitter, A.: Loss-balanced task weighting to reduce negative transfer in multi-task learning. In: Proceedings of the AAAI conference on Artificial Intelligence, vol. 33, pp. 9977–9978 (2019)
41. Liu, X.Q., Liu, X.S.: Markov blanket and markov boundary of multiple variables. J. Mach. Learn. Res. **19**(43), 1–50 (2018)
42. Lugosi, G., Papaspiliopoulos, O., Stoltz, G.: Online multi-task learning with hard constraints. arXiv preprint arXiv:0902.3526 (2009)
43. Mahmood, R., Lucas, J., Alvarez, J.M., Fidler, S., Law, M.: Optimizing data collection for machine learning. Adv. Neural. Inf. Process. Syst. **35**, 29915–29928 (2022)
44. Meng, Z., Yao, X., Sun, L.: Multi-task distillation: towards mitigating the negative transfer in multi-task learning. In: 2021 IEEE International Conference on Image Processing (ICIP), pp. 389–393. IEEE (2021)
45. Moon, C., Kim, D.: Prediction of drug-target interactions through multi-task learning. Sci. Rep. **12**(1), 18323 (2022)
46. Mukherjee, S., Xie, Q., Hanna, J., Nowak, R.: Multi-task representation learning for pure exploration in bilinear bandits. Adv. Neural Inf. Process. Syst. **36** (2024)
47. Pearl, J.: Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan kaufmann (1988)

48. Pearl, J.: Causality. Cambridge university press (2009)
49. Rejwan, I., Mansour, Y.: Top-k combinatorial bandits with full-bandit feedback. In: Algorithmic Learning Theory, pp. 752–776. PMLR (2020)
50. Rosenbaum, L., Dörr, A., Bauer, M.R., Boeckler, F.M., Zell, A.: Inferring multi-target QSAR models with taxonomy-based multi-task learning. J. cheminformatics **5**(1), 1–20 (2013)
51. Rzhetsky, A., Foster, J.G., Foster, I.T., Evans, J.A.: Choosing experiments to accelerate collective discovery. Proc. Natl. Acad. Sci. **112**(47), 14569–14574 (2015)
52. Scarlett, J., Bogunovic, I., Cevher, V.: Overlapping multi-bandit best arm identification. In: 2019 IEEE International Symposium on Information Theory (ISIT), pp. 2544–2548. IEEE (2019)
53. Sener, O., Koltun, V.: Multi-task learning as multi-objective optimization. Adv. neural inf. process. syst. **31** (2018)
54. Sessa, P.G., Laforgue, P., Cesa-Bianchi, N., Krause, A.: Multitask learning with no regret: from improved confidence bounds to active learning. Adv. Neural Inf. Process. Syst. **36**, 6770–6781 (2024)
55. Song, X., Zheng, S., Cao, W., Yu, J., Bian, J.: Efficient and effective multi-task grouping via meta learning on task combinations. Adv. Neural Inf. Process. Syst. **35**, 37647–37659 (2022)
56. Song, Y., Wang, T., Cai, P., Mondal, S.K., Sahoo, J.P.: A comprehensive survey of few-shot learning: Evolution, applications, challenges, and opportunities. ACM Computing Surveys (2023)
57. Unterthiner, T., et al.: Multi-task deep networks for drug target prediction. In: Neural information processing system, vol. 2014, pp. 1–4. NeurIPS (2014)
58. Valsecchi, C., Collarile, M., Grisoni, F., Todeschini, R., Ballabio, D., Consonni, V.: Predicting molecular activity on nuclear receptors by multitask neural networks. J. Chemom. **36**(2), e3325 (2022)
59. Vilalta, R., Drissi, Y.: A perspective view and survey of meta-learning. Artif. Intell. Rev. **18**, 77–95 (2002)
60. Wang, J., Dokholyan, N.V.: Yuel: Improving the generalizability of structure-free compound-protein interaction prediction. J. Chem. Inf. Model. **62**(3), 463–471 (2022)
61. Xin, D., Ghorbani, B., Gilmer, J., Garg, A., Firat, O.: Do current multi-task optimization methods in deep learning even help? Adv. Neural. Inf. Process. Syst. **35**, 13597–13609 (2022)
62. Xu, P., Zhu, X., Clifton, D.A.: Multimodal learning with transformers: a survey. IEEE Trans. Pattern Anal. Mach. Intell. **45**(10), 12113–12132 (2023)
63. Yang, E., Pan, J., Wang, X., Yu, H., Shen, L., Chen, X., Xiao, L., Jiang, J., Guo, G.: Adatask: A task-aware adaptive learning rate approach to multi-task learning. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 37, pp. 10745–10753 (2023)
64. Yu, T., Kumar, S., Gupta, A., Levine, S., Hausman, K., Finn, C.: Gradient surgery for multi-task learning. Adv. Neural. Inf. Process. Syst. **33**, 5824–5836 (2020)
65. Zhou, D., Xu, Z., Li, W., Xie, X., Peng, S.: MultiDTI: drug-target interaction prediction based on multi-modal representation learning to bridge the gap between new chemical entities and known heterogeneous network. Bioinformatics **37**(23), 4485–4492 (2021)

Diagnostic Modeling to Identify Unrecognized Inpatient Hypercapnia Using Health Record Data

Brian W. Locke[1](✉), W. Wayne Richards[1,2], Jeanette P. Brown[1], Wanting Cui[2], Joseph Finkelstein[2], Krishna M. Sundar[1], and Ramkiran Gouripeddi[2,3]

[1] Division of Pulmonary, Critical Care, and Occupational Pulmonary Medicine, University of Utah, Salt Lake City, UT, USA
brian.locke@hsc.utah.edu

[2] Department of Biomedical Informatics, University of Utah, Salt Lake City, UT, USA

[3] Clinical and Translational Science Institute, University of Utah, Salt Lake City, UT, USA

Abstract. Hypercapnic respiratory failure (an accumulation of carbon dioxide, CO_2, in the blood) is often missed in clinical practice. Arterial blood gas is the standard diagnostic test, but it is painful and not routine. When clinicians fail to make the diagnosis, it is often because an arterial blood gas was not obtained. This 'partial verification' of CO_2 levels presents a challenge for machine learning algorithms. We assessed the accuracy of two machine learning methods using demographics and routine lab work to estimate the likelihood that a patient has hypercapnic respiratory failure at hospital admission. Hospitalized patients who received an arterial blood gas sample constituted the training (n = 111,015) and geographic validation (n = 20,834) sets. Acceptance of "silver standard" diagnostic criteria and weighting observations by their modeled likelihood of receiving arterial blood gas sampling were used to assess the stability of findings in the presence of partial verification. Both regularized logistic regression and random-forest-based models resulted in acceptable performance (area under the curve: 0.763 and 0.758 respectively), with minimal changes in the auxiliary analyses. This work suggests that routinely available health record data can stratify the likelihood of hypercapnic respiratory failure among hospitalized adults, and findings may generalize to patients who have not received arterial blood gas sampling in clinical practice.

Keywords: Hypercapnic Respiratory Failure · Diagnostic Model · Machine Learning · Partial Verification

1 Introduction

Hypercapnic respiratory failure is a condition where the amount of inspired air that participates in gas exchange is insufficient to match the metabolic production of carbon dioxide (CO_2) in the body, leading to a buildup of CO_2 in the blood called hypercapnia. Hypercapnic respiratory failure is common among patients presenting to hospitals [1], is associated with high rates of readmission [2], and indicates a high risk of death in the months after recognition [3, 4].

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 36–45, 2024.
https://doi.org/10.1007/978-3-031-66538-7_4

Clinicians often fail to recognize patients who have hypercapnic respiratory failure [5]. Organ failure or death occurs when respiratory failure leads to low oxygen levels, but high blood CO_2 levels are acutely better tolerated. Thus, unlike oxygen, blood CO_2 monitoring is not routine in hospitalized patients. Arterial blood gas sampling is the reference standard diagnostic test to confirm hypercapnia. It is painful [6] and can lead to complications [7], so clinicians only order the test when their suspicion for hypercapnia is particularly high.

Methods to reliably predict which patients have hypercapnic respiratory failure could improve patient outcomes by helping to identify high-risk patients that are missed in clinical practice. For many of these patients, evidence-based treatments exist to improve their symptoms, lessen their risk of hospitalization, and improve their mortality[8]. Some routinely obtained lab values, like serum potassium and bicarbonate, change the likelihood of hypercapnia substantially [9, 10]. Using additional data elements to model the likelihood of hypercapnia being present, termed diagnostic modeling, might improve diagnostic reliability. However, diagnostic models for hypercapnic respiratory failure have not been previously reported.

A key challenge to developing such models is partial verification [11]: blood CO_2 levels are known only for the patients who underwent arterial blood sampling. The patients who would benefit most from more reliable diagnosis are less likely to have received arterial blood gas testing, and are therefore under-represented among the cases available for training a diagnostic model. In the literature evaluating medical tests, two commonly used approaches to address partial verification bias are analyses accepting “silver standard” diagnostic criteria and weighting observations by the likelihood that they’d receive the definitive test. In this work, we aim to assess the accuracy of two approaches to diagnostic modeling of hypercapnic respiratory failure in the presence of partial verification blood CO_2 levels.

2 Methods

This was a retrospective analysis of de-identified electronic health record data from the TriNetX research network database (TriNetX, LLC. Cambridge, Massachusetts) and was exempted by the University of Utah Institutional Review Board (#00152089).

2.1 Data Source

The TriNetX research network is a federated network of electronic health record data from 76 medical centers across the US, serving roughly 115 million patients [12]. All adult, inpatient encounters occurring during the calendar year 2022 that met any of the following criteria were requested: received a diagnostic code for any respiratory failure, had a condition known to cause hypercapnia (including severe obesity), received a procedure code for the treatment of respiratory failure (non-invasive or invasive ventilation), or had an arterial (ABG) or venous blood gas (VBG) obtained on the first day of the encounter. These criteria indicate that consideration of hypercapnia was warranted, which is the spectrum of patients in whom diagnostic modeling would be used [13]. First encounters for each patient were used. Data was cleaned to remove physiologically impossible data and encounters with evidence of incomplete data submission to TriNetX (for example: missing categories of data such as no procedure codes or diagnoses for the encounter of interest).

2.2 Model Creation and Feature Selection

Predictors (age, sex, body mass index [BMI], components of basic blood chemistry testing [sodium, potassium, chloride, bicarbonate, urea, and creatinine], and hemoglobin) were selected a priori on clinical relevance. Additionally, these data elements are routinely present in all hospital admissions and are clinically ascertained independent of (and, generally, before) knowledge of the presence of hypercapnic respiratory failure. No imputation was performed as data elements were selected for low missingness.

To focus on the influence of partial verification, two standard machine learning approaches were used: logistic regression with L1-regularization (termed logistic least absolute shrinkage and selection operator, or LASSO regression, for short) and random forest modeling. LASSO and random forest models were selected for their ubiquity and to balance understandble model outputs compared to possible performance gains from the handling of non-linear relationships. For the LASSO regression, continuous predictors were represented as restricted cubic splines with 4 knots, and 10-fold cross-validation was used to select the minimum prediction error λ. For random forest, hyperparameter (tree depth, number of splitting features, and number of bootstrapped trees) tuning was performed using grid search and 10-fold cross-validation.

2.3 Performance Analysis

Patients who had any ABG on the calendar day of admission showing a partial pressure of CO_2 ($PaCO_2$) over 45 mm Hg were considered to have hypercapnia. In the primary analysis, diagnostic model predictions of the likelihood of hypercapnia were generated for patients in whom ABG sampling was performed. Model discrimination (ability to separate patients with hypercapnia from those without) was assessed using receiver operating characteristic (ROC) curves and summarized as the area under the ROC curve (AUC). Model calibration (how closely the predicted likelihood of hypercapnia correlates with the true likelihood) was assessed by the full sample expected to observed event ratio (E:O), calibration in the large (CITL; the relation of the mean predicted risk to the mean observed risk), calibration slope (CS, slope; whether risks are too extreme for high- and low-risk patients) and calibration plots visualized by decile of predicted risk. Overall performance was summarized using Brier scores. The importance of individual predictors was assessed using standardized regression coefficients (LASSO) and mean decrease in impurity (random forest). Models were trained on data submitted from the Western, Southeastern, and Northeastern US regions, and evaluated in hospitals from the Midwestern region. The evaluation region was chosen arbitrarily by two coin flips and sensitivity analyses holding out the other regions for evaluation show similar results but are omitted for brevity. Geographic validation, rather than a random hold-out set, was performed for a more severe test of the distribution shift associated with regional variations in care practices.

2.4 Evaluation of the Influence of Partial Verification

Two additional analyses were performed to assess the applicability of diagnostic modeling to patients who did not receive reference-standard (ABG) blood CO_2 level assessment. First, VBG sampling is considered a "silver standard" method of diagnosis, with high venous CO_2 levels serving as an imperfect but usable surrogate for arterial CO_2 levels [14]. The same performance metrics were calculated on patients who had either an ABG or VBG that showed hypercapnia (an arterial $PaCO_2$ over 45 mm Hg or a venous PCO_2 over 50 mm Hg) obtained on the day of admission.

Second, inverse probability weighting was used to create a *pseudopopulation* that represents the sample where all patients had an equal probability of receiving ABG sampling, conditional on the covariables used to model the propensity of receiving an ABG sample [11]. Logistic regression was used to generate propensity scores, using prior diagnoses, demographics, lab values, and outpatient medications clinically suspected to relate to the propensity to receive an ABG. The missing indicator method was used to account for missing data [14]. The model was then evaluated in the weighted population to estimate the model's performance if it had been applied across the entire population, as opposed to only those who actually received an ABG.

Statistical analysis used Stata 18 (StataCorp, College Station, TX) with pmcalplot [15] and scikit-learn (1.4.1) [16] via c_ml_stata_cv packages [17].

3 Results

Of 401,079 potentially eligible adult, inpatient encounters, n = 32,987 were excluded for missing data, and n = 68,190 were repeat encounters (Fig. 1). For the primary analysis, an admission-day ABG sample was obtained in 131,849 of 299,902 patients. An additional n = 44,600 had a VBG and were included in the supplementary analysis.

Demographics of the included groups are given in Table 1. Among the patients who underwent ABG sampling on the day of hospital admission, 30% (n = 39,676) had hypercapnia. When VBG verification was accepted to determine hypercapnia status, 33% (n = 57,349) of patients who received either an arterial or venous blood gas showed evidence of hypercapnia. All predictors had $<10\%$ missingness rates.

LASSO logistic regression modeling achieved an AUC of 0.763 and a Brier score of 0.180 in the test set (Midwestern US Hospitals), with adequate calibration (Fig. 2). Serum bicarbonate, potassium, and BMI were the strongest individual predictors. Performance dropped to an AUC of 0.749 and a Brier score of 0.190 in the silver standard analysis and increased to an AUC of 0.792 and a Brier Score of 0.178 when inverse propensity weighting was performed (Table 2).

Random forest-based predictions achieved clinically indistinguishable performance in the primary (AUC 0.758, Brier score 0.184), silver standard (AUC 0.745, Brier Score 0.193), and inverse probability-weighted analysis (AUC 0.782, Brier score 0.184). Particularly in the silver standard analysis, there was a mild underestimation of the likelihood of hypercapnia across all risk categories in both regression and random forest-based models. Serum bicarbonate had the highest feature importance, followed by serum creatinine, hemoglobin, and potassium.

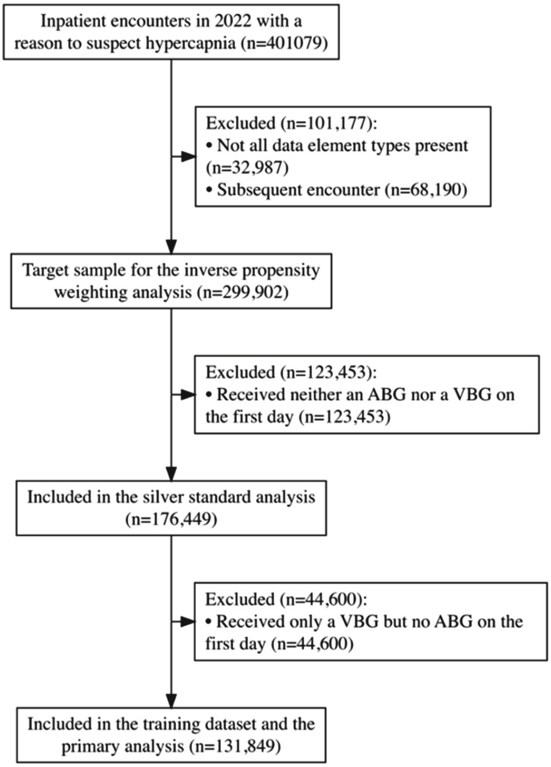

Inpatient encounters in 2022 with a reason to suspect hypercapnia (n=401079)
Excluded (n=101,177):
• Not all data element types present (n=32,987)
• Subsequent encounter (n=68,190)
Target sample for the inverse propensity weighting analysis (n=299,902)
Excluded (n=123,453):
• Received neither an ABG nor a VBG on the first day (n=123,453)
Included in the silver standard analysis (n=176,449)
Excluded (n=44,600):
• Received only a VBG but no ABG on the first day (n=44,600)
Included in the training dataset and the primary analysis (n=131,849)

Fig. 1. Enrollment Flowchart. Only patients who received arterial blood gas sampling to verify their hypercapnia status were included in the primary analysis. ABG = arterial blood gas. VBG = venous blood gas

Table 1. Characteristics of included patients. Western, Northeastern, and Southeastern US hospitals constituted the training set, while Midwestern hospitals were the testing set. Diagnoses were based on diagnosis codes rendered up to the admission of interest. BMI = body mass index, COPD = chronic obstructive pulmonary disease, ABG = arterial blood gas (threshold 45 mmHg), VBG = venous blood gas (threshold 50 mmHg). EHR-recorded death occurred with a median follow-up of 11 months.

	Entire Cohort	First-day ABG or VBG obtained	First-day ABG obtained
		Silver standard analysis	Training, primary analyses
	N = 299,902	N = 176,449	N = 131,849
Training dataset	Not applicable	Not applicable	84% (111,015)
Testing dataset	Not applicable	15% (27,138)	16% (20,834)
Age (years)	62 (±17)	62 (±17)	62 (±17)

(*continued*)

Table 1. (*continued*)

	Entire Cohort	First-day ABG or VBG obtained	First-day ABG obtained
		Silver standard analysis	Training, primary analyses
	N = 299,902	N = 176,449	N = 131,849
Female	47% (141,032)	45% (79,805)	45% (58,792)
Black or African American	18% (53,933)	17% (30,483)	17% (22,006)
Asian	2% (5,866)	2% (3,860)	2% (2,922)
White	68% (204,471)	67% (118,890)	68% (89,414)
Hispanic or Latino Ethnicity	6% (18,366)	6% (10,777)	6% (7,574)
BMI (kg/m^2)	30 (±9)	29 (±8)	29 (±8)
Heart failure	17% (50,053)	16% (27,761)	16% (20,675)
Chronic kidney disease	15% (45,599)	15% (25,670)	14% (18,521)
COPD	15% (45,301)	15% (26,137)	15% (19,309)
Neuromuscular disease	3% (10,338)	3% (6,015)	4% (4,686)
Obstructive sleep apnea	14% (42,029)	11% (20,052)	11% (14,512)
Hypercapnia on admission-day ABG			
No ABG	56% (168,053)	25% (44,600)	0% (0)
All $PCO_2 <$ threshold	31% (92,173)	52% (92,173)	70% (92,173)
$PCO_2 \geq$ threshold	13% (39,676)	22% (39,676)	30% (39,676)
Hypercapnia on admission-day ABG or VBG			
No VBG or ABG	41% (123,439)	0% (0)	0% (0)
All $PCO_2 <$ threshold	40% (119,104)	67% (119,104)	66% (87,129)
Any $PCO_2 \geq$ threshold	19% (57,349)	33% (57,349)	34% (44,722)
Critical care services	26% (78,937)	35% (60,943)	38% (49,917)
Death	16% (47,012)	18% (32,560)	19% (25,646)

Table 2. Performance Metrics. Only test set results are shown. Discrim. = discrimination, ABG = arterial blood gas, VBG = venous blood gas, LASSO = least absolute shrinkage and selection operator, RF = random forest, E:O = expected to observed ratio, CITL = calibration in the large, CS = calibration slope, AUC = area under the receiver operating characteristic curve. A score of 1 is perfect for the AUC, E:O, and CS. For CITL and Brier Score, a score of 0 is perfect.

Analysis	Patients	Model	Discrim	Calibration			Overall
			AUC	E:O	CITL	CS	Brier Score
Primary	ABG	LASSO	0.763	0.986	0.028	1.110	0.180
		RF	0.758	0.969	0.062	1.127	0.184

(*continued*)

Table 2. (*continued*)

Analysis	Patients	Model	Discrim	Calibration			Overall
			AUC	E:O	CITL	CS	Brier Score
Silver Standard	ABG or VBG	LASSO	0.749	0.925	0.156	1.098	0.190
		RF	0.745	0.912	0.180	1.190	0.193
Inverse Probability Weighted	ABG (weighted to full sample)	LASSO	0.792	0.982	0.041	1.227	0.178
		RF	0.782	0.957	0.094	1.251	0.184

4 Discussion

Both LASSO logistic regression and random forest-based diagnostic models achieved acceptable discrimination and calibration for identifying patients with hypercapnic respiratory failure at the time of hospital admission. Both models' performance was maintained when two methods assessing the impact of partial outcome ($PaCO_2$) verification were applied. This suggests that either modeling approach may help identify patients with hypercapnia that are currently not recognized in clinical care.

Both models exclusively rely on predictors collected in nearly all acutely hospitalized patients (routine lab work and demographics). This modeling approach could be used to risk-stratify patients that plausibly have hypercapnia. As can be seen from Fig. 2, panels 1A and 1B, both models were able to render stronger "rule-in" predictions (raising the likelihood of hypercapnia) as compared to "rule-out" predictions, which suggests utility flagging additional patients for further workup as opposed to identifying patients in whom further testing is not indicated..

A key barrier to developing a diagnostic model for hypercapnic respiratory failure is that not all patients receive an assessment of their blood CO_2 levels. To address this, two approaches could be considered. Theoretically, a model could be trained on a prospectively constructed research cohort where all patients undergo ABG assessment. However, the representativeness of patients consenting to participate in this research and the difficulty of enrolling enough patients to train robust machine-learning models limit the feasibility of this approach. Alternatively, models can be trained on large, existing databases, with the acknowledgment that the patients who receive arterial blood gases (and thus are available for model training) may be different from those who do not (and thus stand to benefit most from diagnostic modeling). The primary purpose of the current analyses was to assess how problematic this difference might be.

Performance was relatively maintained when the models were applied to a broader set of patients (those receiving either an ABG or VBG) and the re-weighted population approximating if all eligible patients had been equally likely to receive an ABG. In fact, the model's performance improved in the re-weighted sample, likely because it is easier to detect compensated hypercapnia (defined as hypercapnia where the kidneys have been able to retain bicarbonate) while those patients are also less likely to undergo ABG sampling due to subtler symptoms.

The findings of this study provide some preliminary reassurance that much of the variability in the type of patients who receive ABG verification of hypercapnia status

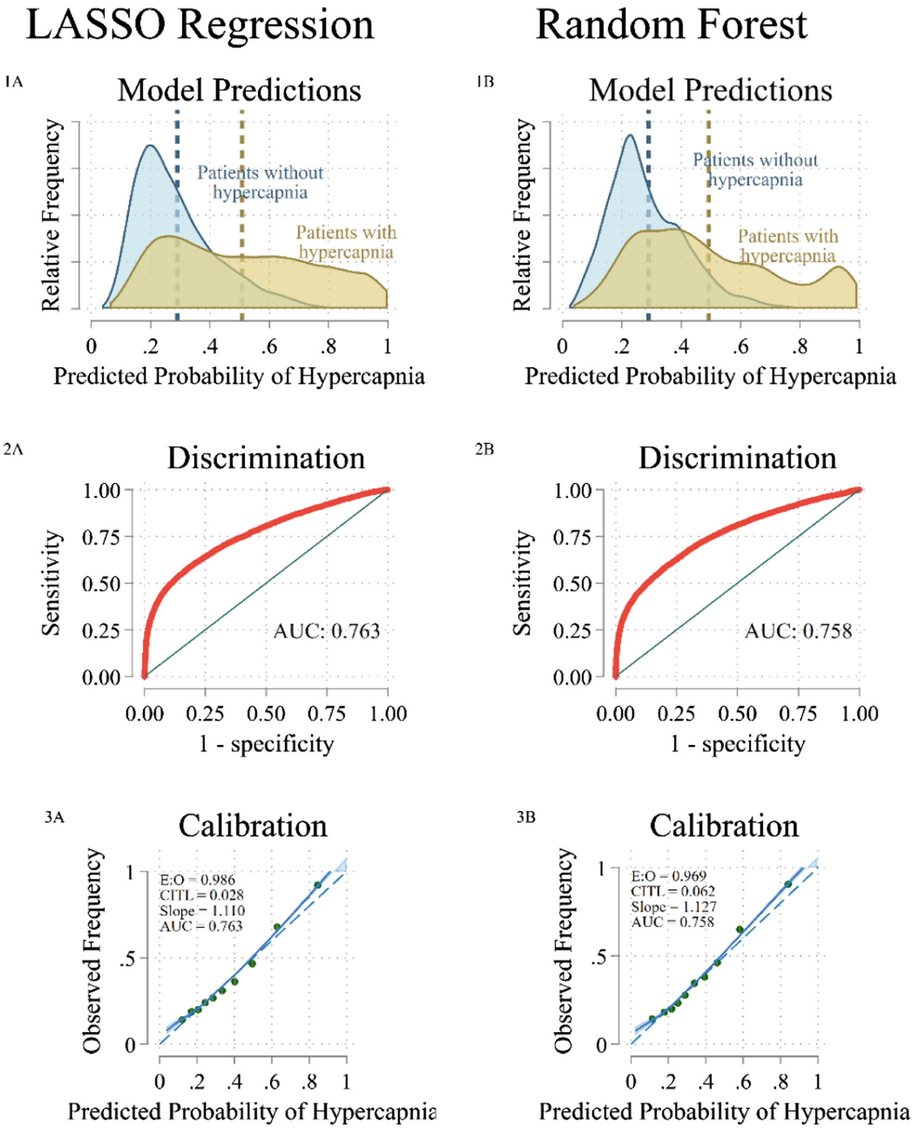


Fig. 2. Performance of L1-regularized (LASSO) logistic regression and random forest-based diagnostic models of the likelihood of hypercapnia at hospital admission. Panel 1A and 1B show the distribution of model predictions by hypercapnia status. Mean predictions are indicated by dashed lines. In panel 3A and 3B, calibration is presented by decile of predicted risk. E:O = expected to observed ratio, CITL = calibration in the large (intercept), slope = calibration slope, AUC = area under the receiver operating characteristic curve.

does not importantly confound the relationship between model predictions and the true likelihood of hypercapnia. Ultimately, however, prospective validation of the model predictions by assessing CO_2 levels in patients who have not received ABG sampling will be required before clinical or research use is advisable.

A notable strength of the study is the large (over 100,000 patients in the training set) and geographically diverse sample, which guards against overfitting and modeling of local, idiosyncratic practice patterns. Only near-universally available predictors from the same hospitalization were used in the model to minimize the influence of informed presence bias and the dependence on inter-institution data linkages.

Several additional limitations exist. First, we used relatively simple machine learning approaches to estimate the generalizability of diagnostic predictions to patients who did not receive arterial blood gas outcome sampling, but more advanced methods of imputation, feature selection, and model choice may improve performance. Though no benchmark for comparison exists, neither model is likely sufficient for stand-alone diagnosis or labeling at the current accuracy. The inclusion of unstructured elements (e.g. signs and symptoms) might improve performance enough for this use. The validity of inverse probability weighting analyses depends on several assumptions that do not strictly hold, though the approximation of performance may still be useful. Similarly, we cannot quantify how much of the performance drop when including venous CO_2 as an outcome occurs due to the imperfect relationship of venous to arterial CO_2. Lastly, predictors and outcomes were matched only to the calendar day, and thus transient changes in blood CO_2 levels may be misclassified. However, actionable long-term treatments for hypercapnic respiratory failure require the persistence of hypercapnia, so the performance in patients with stable elevations may be more clinically relevant.

In summary, we show that diagnostic modeling of the likelihood of hypercapnia using routine lab and demographic data is likely sufficient for risk stratification and may perform well on patients who do not currently receive definitive ABG diagnosis.

Acknowledgments. This research was supported by the National Institutes of Health under Ruth L. Kirschstein National Research Service Award 5T32HL105321 from the NIH. (B.W.L.), the American Thoracic Society Academic Sleep Pulmonary Integrated Research/Clinical Fellowship (ASPIRE) Fellowship, UL1TR002538, and UM1TR004409 awards from NIH NCATS (R.G.). The data used in this study was collected on June 2, 2023 from the TriNetX Research Network.

Disclosure of Interests. K.M.S. Sundar is co-founder of Hypnoscure LLC—a software application for population management of sleep apnea through the University of Utah Technology Commercialization Office. J.F. is a general chair for the AIME 2024 conference. All other authors report no conflict of interest.

References

1. Chung, Y., Garden, F.L., Marks, G.B., Vedam, H.: Population prevalence of hypercapnic respiratory failure from any cause. Am. J. Respir. Crit. Care Med. (2022). https://doi.org/10.1164/rccm.202108-1912le

2. Meservey, A.J., Burton, M.C., Priest, J.S., Teneback, C.C., Dixon, A.E.: Risk of readmission and mortality following hospitalization with hypercapnic respiratory failure. Lung (2020). https://doi.org/10.1007/s00408-019-00300-w
3. Wilson, M.W., Labaki, W.W., Choi, P.J.: Mortality and healthcare utilization of patients with compensated hypercapnia. Ann. Am. Thorac. Soc. (2021). https://doi.org/10.1513/annalsats.202009-1197oc
4. Vonderbank, S., et al.: Hypercapnia at hospital admission as a predictor of mortality. Open Access Emerg. Med. OAEM **12**, 173–180 (2020). https://doi.org/10.2147/OAEM.S242075
5. Nowbar, S., et al.: Obesity-associated hypoventilation in hospitalized patients: prevalence, effects, and outcome. Am. J. Med. (2004). https://doi.org/10.1016/j.amjmed.2003.08.022
6. Gonella, S., et al.: Interventions to reduce arterial puncture-related pain: a systematic review and meta-analysis. Int. J. Nurs. Stud. (2021). https://doi.org/10.1016/j.ijnurstu.2021.104131
7. Rowling, S.C., Fløjstrup, M., Henriksen, D.P., et al.: Arterial blood gas analysis: as safe as we think? A multicentre historical cohort study. ERJ Open Res. (2022). https://doi.org/10.1183/23120541.00535-2021
8. Gay, P.C., Owens, R.L.: Executive summary: optimal NIV medicare access promotion: a technical expert panel report from the american college of chest physicians, the american association for respiratory care, the american academy of sleep medicine, and the American thoracic society. Chest **160**, 1808–1821 (2021). https://doi.org/10.1016/j.chest.2021.05.074
9. Mokhlesi, B., et al.: Evaluation and management of obesity hypoventilation syndrome. An official American thoracic society clinical practice guideline. Am. J. Respir. Crit. Care Med. (2019). https://doi.org/10.1164/rccm.201905-1071st
10. Locke, B., Gouripeddi, R., Richards, W., Brown, J., Sundar, K.: Test performance of serum bicarbonate in identifying hypercapnia across settings and diseases. In: D30. Integrating OSA and Comorbidities for Effective Therapies, pp. A6495–A6495. American Thoracic Society (2023)
11. Pepe, M.S.: The Statistical Evaluation of Medical Tests for Classification and Prediction. Oxford University Press (2003). https://doi.org/10.1198/tech.2005.s278
12. Palchuk, M., et al.: A global federated real-world data and analytics platform for research. JAMIA Open (2023). https://doi.org/10.1093/jamiaopen/ooad035
13. Usher-Smith, J.A., Sharp, S.J., Griffin, S.J.: The spectrum effect in tests for risk prediction, screening, and diagnosis. BMJ **353** (2016). https://doi.org/10.1136/bmj.i3139
14. Groenwold, R.H., White, I.R., Donders, A.R.T., Carpenter, J.R., Altman, D.G., Moons, K.G.: Missing covariate data in clinical research: when and when not to use the missing-indicator method for analysis. CMAJ **184**, 1265–1269 (2012). https://doi.org/10.1503/cmaj.110977
15. Ensor, J., Snell, K.IE., Martin, E.C.: PMCALPLOT: stata module to produce calibration plot of prediction model performance (2018). https://ideas.repec.org/c/boc/bocode/s458486.html
16. Pedregosa, F., et al.: Scikit-learn: machine learning in Python. J. Mach. Learn. Res. **12**, 2825–2830 (2011)
17. Cerulli, G.: Machine learning using Stata/Python. Stata J. **22**, 772–810 (2022). https://doi.org/10.1177/1536867X221140944

Enhancing Hypotension Prediction in Real-Time Patient Monitoring Through Deep Learning: A Novel Application of XResNet with Contrastive Learning and Value Attention Mechanisms

Xiangru Chen and Milos Hauskrecht(✉)

Computer Science Department, University of Pittsburgh, Pittsburgh, PA 15260, USA
{xic197,milos}@pitt.edu

Abstract. The precise prediction of hypotension is vital for advancing preemptive patient care strategies. Traditional machine learning approaches, while instrumental in this field, are hampered by their dependence on structured historical data and manual feature extraction techniques. These methods often fall short of recognizing the intricate patterns present in physiological signals. Addressing this limitation, our study introduces an innovative application of deep learning technologies, utilizing a sophisticated end-to-end architecture grounded in XResNet. This architecture is further enhanced by the integration of contrastive learning and a value attention mechanism, specifically tailored to analyze arterial blood pressure (ABP) waveform signals. Our approach improves the performance of hypotension prediction over the existing state-of-the-art ABP model [7]. This research represents a step towards optimizing patient care, embodying the next generation of AI-driven healthcare solutions. Through our findings, we demonstrate the promise of deep learning in overcoming the limitations of conventional prediction models, thereby offering an avenue for enhancing patient outcomes in clinical settings.

Keywords: Hypotension prediction · Contrastive learning · Real-time monitoring

1 Introduction

Accurately predicting hypotension is crucial for effective patient care and management, particularly during surgical procedures. Traditional machine learning methods, known for their interpretability and ease of implementation, have effectively utilized structured historical clinical data and manual feature extraction. However, these may not capture the dynamic nature of physiological signals preceding a hypotensive event.

Deep learning, which excels at automatically learning data representations without manual feature engineering, offers a robust solution for analyzing time-series data such as ABP waveform signals. Our hypothesis suggests that these

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 46–51, 2024.
https://doi.org/10.1007/978-3-031-66538-7_5

signals contain early indicators of cardiovascular instability, potentially leading to hypotensive episodes. By leveraging deep learning to analyze ABP signals, we aim to enhance the accuracy and timeliness of hypotension predictions, potentially uncovering new predictive biomarkers.

In this paper, we utilize a modern time-series analysis architecture based on XResNet, and extend it with a value attention structure to further improve its performance and with contrastive learning for model training. This approach significantly enhances the prediction of hypotension through ABP signals from VitalDB [6] dataset when compared to the existing baseline models, showcasing the potential of deep learning technologies in transforming real-time patient monitoring and care. This integration marks a step forward in the application of AI-driven medical prediction and intervention strategies.

2 Related Work

Various methods for detecting hypotension exist, yet real-time monitoring capabilities are insufficient. Many studies significantly manipulate their datasets by either removing many samples through strict criteria or discarding normal blood pressure readings to balance data. This approach deviates from the goal of real-time hypotension prediction, where normal conditions prevail and should be preserved. Effective real-time monitoring needs to include the full range of blood pressure data to accurately reflect patient conditions. Over-simplifying datasets for theoretical accuracy does not meet the urgent need for hypotension detection in clinical settings. Predictive models should be developed and validated in realistic scenarios, emphasizing the ability to process and interpret data promptly for practical and effective clinical use.

3 Methodology

Our model for hypotension prediction applies XResNet architecture for time-series data [8,9] to ABP waveform data. We further enhance the architecture with the attention layer able to integrate absolute blood pressure signals into the waveform models. Finally, to improve the model learning we explore temporal and contextual contrasting (TS-TCC) [5], a time-series contrastive learning framework to pre-train the models.

Intuitively, ABP signal's value informs the XResNet model what is the baseline blood pressure and how far it is from the hypotension region. We designed a simple module external to XResNet that computes the average value of the blood pressure signals and is concatenated with the output of XResNet. This approach enhances the model's ability to prioritize and weigh blood pressure readings. The modification is shown in Fig. 1.

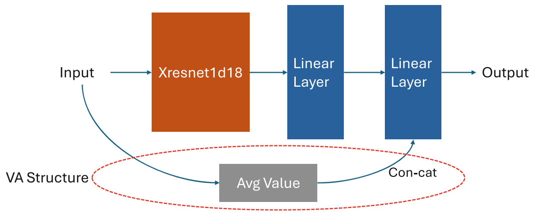


Fig. 1. Our Xresnet architecture with added value attention (VA) structure.

4 Experiments

4.1 Data and Preparation Strategies

We utilized arterial blood pressure (ABP) signals from the public VitalDB [6], which contains BP recordings from surgeries. Records from 3,458 patients, each spanning several hours, were split into training, testing, and validation sets with a ratio of 6:2:2, using 30-s samples. Hypotension events are defined as readings under 90 mm Hg systolic or 60 mm Hg diastolic for over a minute.

Our goal is to predict hypotensive events 5–10 min before they occur. To optimize training, we excluded data from 0–5 min prior and 10–20 min post-event from both training and validation datasets. We developed two test strategies for real-time monitoring conditions. Test strategy 1 predicts hypotension within the next 5–10 min based solely on past and present data, excluding predictions during ongoing hypotension or during the immediate recovery. Test strategy 2 mirrors the training set approach, where false positives may indicate either a late prediction or early hypotension signs.

4.2 Baseline and Metrics

We conducted a comparison between our methods and a residual neural network structure proposed earlier by Jo, Y et al. [7], which achieved a good predictive performance on the hypotension prediction task with the same data source. We note that past results for the method were obtained on a biased test dataset with a balanced set of positive and negative instances, a common practice in previous studies, that, however, does not align with priors observed in real-time monitoring applications. To make the comparison we used ROC curve (AUROC), AUPRC (Area Under the Precision-Recall Curve) and the Precision-Recall (P-R) curve as metrics, while considering the extreme class imbalance typical of real-time medical monitoring data.

4.3 Results

Results in Tables 1 and 2 show that our model architectures are superior to the baseline model in terms of AUROC and AUPRC. Notably, the best results were achieved using the XResNet combined with TS-TCC and the VA Structure. The

Table 1. Performance Comparison on Test Strategy 1.

Model	AUROC	AUPRC
Baseline	0.6990	0.1723
XResNet18	0.7114	0.1811
XResNet18+TS-TCC	0.7147	0.1836
XResNet18+VA Structure	0.7059	0.1788
XResNet18+TS-TCC+VA Structure	**0.7194**	**0.1876**

Table 2. Performance Comparison on Test Strategy 2.

Model	AUROC	AUPRC
Baseline	0.7080	0.1885
XResNet18	0.7208	0.2019
XResNet18+TS-TCC	0.7240	0.2016
XResNet18+VA Structure	0.7149	0.1964
XResNet18+TS-TCC+VA Structure	**0.7297**	**0.2065**

TS-TCC uses scaling as data augmentation, which leads to the XResNet model losing some focus on the absolute values of ABP. Our VA Structure effectively mitigated this drawback by enhancing the model's sensitivity to the numerical values. Additionally, we conducted a comparative analysis of Precision-Recall (PR) curves between the baseline and our model that combines XResNet18 with TS-TCC and VA Structure. Figures 2 and 3 reveal that, at recall rates below 0.5, the precision of our method consistently exceeds that of the baseline. This further substantiates the enhancement our approach offers in the context of real-time monitoring, showcasing its superior performance in predicting events within the specified recall threshold.

As anticipated, the overall AUPRC and PR curve metrics were somewhat lower. This can be attributed to our 'real-time' monitoring test setup, which leads to an unbalanced test set with positive instance prior of 0.084 for test strategy 1 and 0.086 for test strategy 2. As expected, the imbalance made the task of accurately identifying positive cases more challenging. Nevertheless, the results show that it is indeed possible to predict future intra-operative hypotension events from Arterial Blood Pressure waveforms signals at reasonable precision. For example our best method was able to achieve the precision of 0.2 at approximately 30% coverage of future hypotension events which is very promising. This warrants further development and exploration of the methodology.

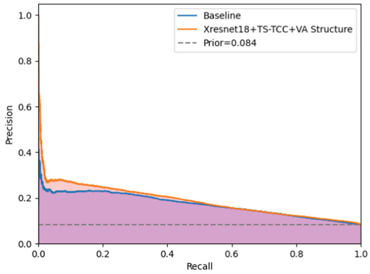


Fig. 2. P-R curve comparison between baseline and our method (XResNet18+TS-TCC+VA Structure) on test strategy 1.

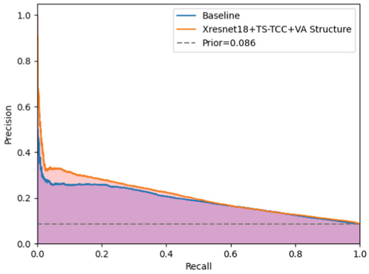


Fig. 3. P-R curve comparison between baseline and our method (XResNet18+TS-TCC+VA Structure) on test strategy 2.

5 Conclusion

Our research presents a new solution to hypotension prediction that relies only on ABP waveform signal and integrates XResNet with TS-TCC and a novel VA structure. This hybrid approach compensates for the diminished focus on crucial blood pressure values during data augmentation, thereby enhancing the model's predictive capabilities. Despite the challenges posed by an unbalanced dataset reflective of real-time monitoring conditions, our models showed initial promising results in predicting future hypotension events from the history of ABP measurements. In addition our models were able to consistently outperform the current SOTA waveform-based hypotension prediction baseline. All this marks an important step forward in the application of deep learning technologies for real-time patient monitoring and hypotension prediction.

Acknowledgments. The work presented in this paper was supported by NIH grant R01-DK131586 and University of Pittsburgh graduate student fellowship. The content of this paper is solely the responsibility of the authors and does not necessarily represent the official views of NIH.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Kim, Y., Seo, J., O'Reilly, U.M.: Large-scale methodological comparison of acute hypotensive episode forecasting using MIMIC2 physiological waveforms. In: IEEE 27th International Symposium on Computer-Based Medical Systems. New York, NY (2014)
2. Zhang, G., Yuan, J., Yu, M., Wu, T., Luo, X., Chen, F.: A machine learning method for acute hypotensive episodes prediction using only non-invasive parameters. Comput. Methods Prog. Biomed. **200**, 105845 (2021)
3. Afsar F. Prediction of acute hypotension episodes in patients taking pressor medication using modeling of arterial blood pressure waveforms. In: 4th International Conference on Bioinformatics and Biomedical Engineering. Chengdu (2010)
4. Lee, J., Woo, J., Kang, A.R., Jeong, Y.S., Jung, W., Lee, M., et al.: Comparative analysis on machine learning and deep learning to predict post-induction hypotension. Sensors. **20**, 4575 (2020)
5. Eldele, E., et al.: Time-series representation learning via temporal and contextual contrasting. arXiv preprint arXiv:2106.14112 (2021)
6. Lee, H.-C., Park, Y., Yoon, S.B., Yang, S.M., Park, D., Jung, C.-W.: VitalDB, a high-fidelity multi-parameter vital signs database in surgical patients. Sci Data. **9**, 1–9 (2022)
7. Jo, Y.Y., et al.: Predicting intraoperative hypotension using deep learning with waveforms of arterial blood pressure, electroencephalogram, and electrocardiogram: Retrospective study. PLoS ONE **17**, e0272055 (2022)
8. He, K.M., Zhang, X.Y., Ren, S.Q., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)

9. He, T., Zhang, Z., Zhang, H., Zhang, Z., Xie, J., Li, M.: Bag of tricks for image classification with convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Computer Vision Foundation/IEEE, pp. 558–567 (2019)
10. Howard, J., Gugger, S.: Fastai: a layered API for deep learning. Information **11**, 108 (2020)
11. Oguiza, I.: "tsai - a state-of-the-art deep learning library for time series and sequential dat", Github (2020). https://github.com/timeseriesAI/tsai
12. Liu, Z., Alavi, A., Li, M., Zhang, X.: Self-supervised contrastive learning for medical time series: a systematic review. Sensors **23**(9), 4221 (2023)

Evaluating the TMR Model for Multimorbidity Decision Support Using a Community-of-Practice Based Methodology

Josip Grgurić, Annette ten Teije(✉), and Frank van Harmelen(✉)

Vrije Univeriteit Amsterdam, Amsterdam, The Netherlands
{annette.ten.teije,frank.van.harmelen}@vu.nl

Abstract. Clinical practice guidelines are typically designed for treatment of a single disease, ignoring undesired interactions for comorbid patients. A number of methods for detecting such guideline interactions have been developed, based on computer interpretable representations of guidelines. A recently published paper by Van Woensel et al. [7] compared a number of methods for detecting and resolving interactions between multiple guidelines. The current paper contributes to this comparative corpus by applying the same functional features and evaluation dimensions to the TMR method for multimorbidity decision support. Our comparison shows that TMR allows for more complex reasoning compared to some of the methods discussed in [7]. It is one of the few that supports automated detection of adverse interactions. However, it falls short on temporal reasoning and reasoning about drug dosage. Our study also represents the first independent validation of the evaluation methodology published in [7].

1 Introduction

Clinical guidelines are crucial for ensuring high-quality medical care as they provide evidence-based recommendations for diagnosing, treating, and managing health conditions. By synthesising the latest research findings and expert consensus, these guidelines offer standardised protocols that help healthcare professionals make informed decisions, reduce variations in practice, and improve patient outcomes. Additionally, guidelines streamline decision-making processes, enhance efficiency in healthcare delivery, and promote patient safety by minimising errors and adverse events.

Treating patients with multimorbidities presents significant challenges for clinical guidelines. Firstly, guidelines typically focus on single diseases or conditions, which may not adequately address the complex interactions and overlapping symptoms present in patients with multiple comorbidities. This can result in conflicting recommendations or difficulties in prioritizing treatments when managing multiple conditions simultaneously. The sheer number of guidelines and recommendations for each individual condition, multiplied by the number of simultaneous conditions a patient might suffer from, can overwhelm clinicians

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 52–63, 2024.
https://doi.org/10.1007/978-3-031-66538-7_6

and complicate decision-making when managing multiple conditions in a single patient. Overall, the complexity and heterogeneity of comorbid patients present significant barriers to effective treatment adherence.

Computer interpretable guidelines (CIGs) can help in managing guideline-based care for comorbid patients. Such CIGs can help identify potential interactions between treatments for different conditions, alerting clinicians to potential adverse effects or contraindications.

A number of CIG models have been developed over the years. A recent survey paper [7] has evaluated the functionality of five of these CIG models on their ability to represent and detect interactions between multiple guidelines when applied to comorbid patients. In this paper, we add the Transitions Based Recommendations (TMR) method to this corpus, thereby extending the insights gained in [7]. As we will show, the TMR method adds a significantly different member to the landscape of CIG models that were studied in [7].

Our study also represents the first independent validation of the evaluation methodology published in [7].

In Sect. 2 of this paper we will summarize the TMR method for modelling guidelines, in Sect. 3 we summarize the evaluation methodology from [7], in Sect. 4 we apply this methodology to the TMR model, and in section compare TMR with other SIG models. Section 6 concludes.

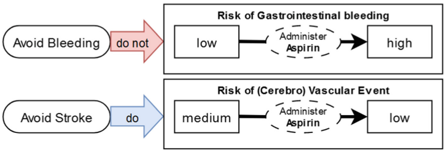
2 TMR

This section summarizes the Transitions Based Recommendations (TMR) model for computer interpretable guidelines, as published in [8,9].

2.1 Core TMR Concepts

The central idea of TMR is to model medical interventions as state transitions, hence the name "Transitions Based Recommendations". The state of a patient can be described through a number of atomic state descriptors, for example "the patient is at high risk of a cerebro-vascular event", and a specific care action, such as "administer dipyridamole", will cause a transition of the patient state to become "at low risk of cerebro-vascular event".

The interventions prescribed in a medical guideline are then modelled as a collection of such state transitions (state → transition → state). Each state transition is labelled with the goal it is aimed to achieve. The second transition states to administer Aspirin in order to avoid a stroke, which transitions the patient from a medium-risk of a cerebro-vascular event to a low risk. The first transition states a contra-indication: in order to avoid bleeding, do not administer aspirin since it will transition the patient from a low risk of gastrointestinal bleeding to a high risk thereof. These examples illustrate the four basic concepts used within TMR:



Contradiction Interactions	**two recommendations that should not be both followed at the same time**
1. Opposed recommendations to the same care action	*- Do not administer aspirin to avoid increasing the risk of gastrointestinal bleeding* *- Administer aspirin to handle inflammation*
2. Opposed recommendations to similar transitions	*- Do not administer beta-blockers to avoid lowering blood pressure* *- Administer ACE inhibitor to lower blood pressure*
3. Recommendations to inverse transitions	*- Administer ACE inhibitor to lower blood pressure* *- Administer midodrine to increase blood pressure*
Repetition Interactions	**set of recommendations that are susceptible to optimization**
4. Repeated recommendations to the same care action	*- Administer aspirin to reduce the risk of thrombus* *- Administer aspirin to relief pain* *- Administer aspirin to handle inflammation*
Alternative Interactions	**set of recommendations that hold as alternatives**
5. Repeated recommendations to the similar transitions promoted by different care action	*- Administer aspirin to handle inflammation* *- Administer ibuprofen to handle inflammation* *- Administer naproxen to handle inflammation*
6. Non-recommended transition whose inverse transition is recommended	*- Do not administer aspirin to avoid increasing the risk of gastrointestinal bleeding* *- Administer PPI to decrease risk of gastrointestinal bleeding*
External Interactions	**interactions detected from external knowledge bases**
7. Alternative drug	*- Administer ibuprofen as alternative to aspirin*
8. Incompatible drug	*- ibuprofen and aspirine are incompatible drugs*

A **situation**, which defines the state of a patient, with the pre-situation before the transition ("low risk of gastrointestinal bleeding") and the post-situation after the transition ("high risk of gastrointestinal bleeding).

A **transition** is the possibility of shifting from one situation to another ("administer aspirin").

A **care action** defines an action which can be taken. Such a care action causes a transition ("avoid bleeding")

A **recommendation** defines whether or not a transition should be pursued or avoided

TMR's state-transition model allows for the detection of interacting treatments in multimorbidity decision support as investigated in [7]. TMR can detect six so-called internal interaction types. These interaction types are called internal because they are triggered internally within or between guidelines. TMR can also detect two external interaction types, which are triggered by the combination of guidelines with external knowledge sources such as Drugbank or SIDER. The table below shows these eight interaction types together with an example for each of them. [8] further elaborates on the conceptual model of TMR.

A powerful feature of TMR is that each of these interaction types is defined by a logical rule. Once a set of medical guidelines has been formalised as TMR state-transitions, these rules can be applied to the set of state-transitions to automatically detect interactions. This is done by analysing only the formal model of the guideline, and no connection to patient data is required. As an example, we give the rule for detecting contradicting recommendations due to similar transitions (interaction type 2 above):

$$\begin{aligned}
&\forall g : \mathit{Guideline}, r_1, r_2 : \mathit{Recommendation}, t_1, t_2 : \mathit{Transition}\\
&(\mathit{partOf}(r_1, g) \wedge \mathit{partOf}(r_2, g) \wedge \mathit{similarTo}(t_1, t_2) \wedge\\
&\mathit{recommendsToPursue}(r_1, t_1) \wedge \mathit{recommendsToAvoid}(r_2, t_2))\\
&\rightarrow \exists i\ (\mathit{ContradictionDueToSimilarTransition}(i) \wedge\\
&\qquad \mathit{relates}(i, r_1) \wedge \mathit{relates}(i, r_2))
\end{aligned}$$

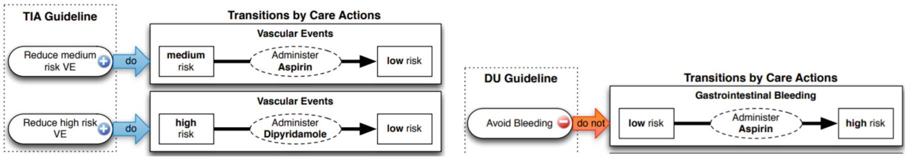
This rule states that if a guideline g contains two recommendations r_1 and r_2, with r_1 recommending to pursue a transition t_1, but r_2 recommends to avoid a transition t_2, while t_1 and t_2 are actually similar transitions, we have a case of a contradiction due to similar opposing recommendations for similar transitions (type 2 above). Similar rules exist for all the other interaction types. [9] shows how the entire ruleset can be efficiently implemented in Prolog, with the external interactions implemented as SPARQL queries to external knowledge graphs.

2.2 Simplified Example of Interaction Detection in TMR

In this subsection we give a simplified example of TMR's mechanism for interaction detection that we described in general terms in the previous subsection. We take as our example the combination of the Duodenal Ulcer (DU) guideline with the guideline for Transient Ischemic Attack (TIA), which is also used as one of the test cases in [7]. The TIA guideline contains the excerpt:

> *"Aspirin (50-325mg/d) monotherapy (Class I; Level of Evidence A) or the combination of 25mg and extended-release dipyridamole 200mg twice daily (Class I; Level of Evidence B) is indicated as initial therapy of TIA or ischemic stroke for prevention of future stroke"*

which (together with some additional context from the guideline text) is modelled as the two TMR state transitions in Fig. 1(a).

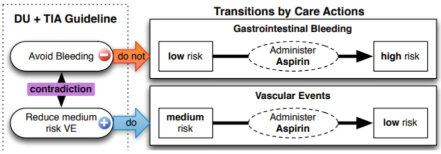


(a) (b)

Fig. 1. Fragments of the TIA Guideline (a) and the DU Guideline (b) in TMR.

A fragment from the DU guideline reads:

> *"29. In patients with low-dose aspirin-associated bleeding ulcers, the need for aspirin should be assessed. If given for secondary prevention (i.e., established cardiovascular disease) then aspirin should be resumed as soon as possible after bleeding ceases in most patients: ideally with 1-3 days and certainly within 7 days."*



The first sentence of this fragment leads to the TMR transition in Fig. 1(b). (The second sentence would have to be modelled in a separate state-transition). The combination of these transitions will trigger the logical rule for "ContradictionDueToSimilarTransition" given above, signalling a conflict between the "do Administer Aspirin" recommendation in the TIA guideline and the "do not Administer Aspirin" recommendation in the DU guideline. Because these rules are specified in a declarative logic, the result of the TMR analysis of side effects is independent of the order in which these guideline recommendations are analysed.

3 Evaluation Methodology

We now summarise the evaluation methodology from [7], before applying this methodology to the TMR model. Building on earlier work in [26], [7] distinguishes 18 functional features that are needed to solve the multimorbidity problem, derived from a literature review and validated by clinicians. These functional features are divided into three groups:

Based on these functional features of CIG-models, [7] defines four evaluation dimensions on which to evaluate the different CIG-models:

DET dimension: Automated detection of adverse interactions between guidelines The DET dimension characterises the automated detection of multimorbidity interactions (e.g., drug-drug, drug-condition) and conflicts (e.g., starting and stopping the same drug). Hence, it covers functional features related to detection (A1-A7 in the table above).

Detection features	Mitigation features	Others
A1 Drug from a CG has an effect on a comorbid condition	B1 Adding a drug to mitigate an ADE	C1 Patient preferences and/or patient burden
A2 Two or more drugs from different CGs may interact	B2 Adjust drug dosage	C2 Optimization of clinical resources
A3 Clinical goals from different CGs may conflict	B3 Monitor the effect of a drug	C3 explanation of the mitigation strategy(ies)
A4 Conflicting actions from different CGs	B4 Replacing a drug with a safer or non-interacting or a more effective drug for comorbidity	C4 Alternative mitigation strategies for a single interaction
A5 Duplicate or redundant advice from different CGs	B5 Discard unsafe or interacting drug	
A6 Temporal relationship between different CGs	B6 Delay a task to avoid a temporal overlap	
A7 Multiple interactions from different CPGs occuring at the same time	B7 Add a task to ensure a temporal overlap	

STRAT dimension: Representation of conflict management strategies Any method to detect and resolve multimorbidity conflicts applies management strategies to detect and/or mitigate conflicts and adverse interactions for a given multimorbidity. The STRAT dimension characterizes the representation of these management strategies, and thus covers detection and mitigation features (A1-A7, B1-B8 in the table above).

IMPL dimension: Implementation paradigm The mitigation of conflicts often reuse or tailor automated planning, graph-based or logical reasoning paradigms. The IMPL dimension characterizes the utilized implementation paradigm, and covers all mitigation features (B1-B8 above).

HUM dimension: Human in the loop mitigation support (HUM) Some methods support interaction with clinicians to help find treatment plans; explanations can be provided to support these interactions. The HUM dimension characterizes these human-in-the-loop aspects, and covers detection and mitigation features (A1-A7, B1-B8 in the table above).

[7] also defines four realistic case studies compiled by the CIG community and validated by clinicians:

Case study 1: Transient Ischemic A + Duodenal ulcer + Osteoporosis
Case study 2: Chronic kidney disease + hypertension+ atrial fibrillation
Case study 3: Venous Thromboembolism + Urinary tract infection
Case study 4: Drug-eluting Stent + lung mass surgery

These different use cases have been chosen so as to fully cover all functional features, as shown in the table below We refer to appendix A of [7] for a detailed description of these case studies.

Case Number	Diseases	Functional Features covered
Case 1	TIA/DU/Osteoporosis	[A1, A4, A7, B1, B4, B5, C1, C3, C4]
Case 2	CKD/HTN/AF	[A1, A2, B4, B5, C1, C2]
Case 3	VTE/UTI	[A2, A6, B2, B3, C1]
Case 4	Stent/Surgery	[A1, A3, A5-A7, B2, B4, B5-B7, C1, C3, C4]

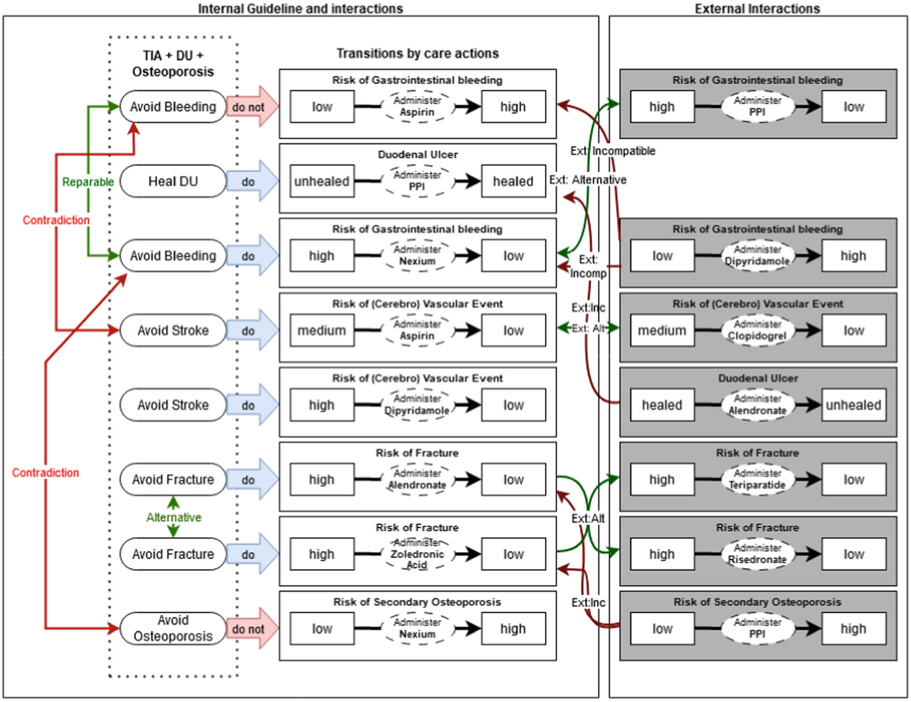
[7] applies this apparatus of functional features, evaluation dimensions and case studies to 6 different CIG models to evaluate to what extent they are able to handle conflicts in each of the 4 multimorbidity case studies. These CIG models are SDA [5], GLARE-SSCPM [4], PROforma-CMM [2], GoCom [1], MitPlan [3] and CigIntO [6]. In the next section, we describe our work on applying this evaluation apparatus from [7] to the TMR model that we described in Sect. 2.

4 Experiments

4.1 Description of the Experiments

We have modelled three of the four use cases of [7] as TMR state-transitions. It was not possible to model case 3 (the VTE/UTI guidelines) in TMR because it requires functional feature B3, which is linked with drug dosage which is not representable in TMR. However, all the other features of case 3 are also covered by the other three cases, so the loss of dropping case 3 is only limited to feature B3. Of the remaining three, we present case 1 (the TIA/DU/Osteoporosis guidelines), the others are in supplementary material[1].

[1] https://cs.vu.nl/~frankh/spool/AIME2024-supplementary-material/.



This model includes the contradiction between administering Aspirin (to avoid stroke) and not administering Aspirin (to avoid bleeding) described in Sect. 2, as well as a similar conflict between administering Nexium (to avoid bleeding) and not administering Nexium (to avoid osteoporosis). Besides these internal interactions (i.e. interactions derivable solely from the text of the guidelines), the diagram also shows a number of external interactions, for example the fact that Clopidogrel is an alternative treatment to Aspirin for reducing the risk of a cerebrovascular event (a fact derived from DrugBank).

4.2 Scoring of TMR on the Functional Features

The following table shows which functional features TMR can represent in all three cases. We will discuss each of the features of case 1.

	A1	A2	A3	A4	A5	A6	A7	B1	B2	B3	B4	B5	B6	B7	C1	C2	C3	C4
Case 1	+			+			-	+			+	-			-		±	+
Case 2	+	+									+	-			-	-		
Case 4	+		+		+	-	-		-		+	-	-	-	-		±	+

+A1: Drug from a CPG has an effect on a comorbid condition: Applying TMR's interaction rules, TMR detects that there exist a contradiction due to same action: administer Aspirin to avoid a risk of stroke, avoid administering aspirin to prevent internal bleeding (see the model for case 1).

+A4: Conflicting actions from different CPGs This feature is implemented within TMR in the exact same way as feature A1. In short, Contradiction due to same action: For case 1 Aspirin is recommended to both avoid and pursue.

-A7: Multiple interactions from different CPGs interacting at the same time The current version of TMR is incapable of dealing with temporal aspects.

+B1: Adding a drug to mitigate an ADE This feature is visible in the model for case 1 with the negative side effects of Aspirin being counteracted by Nexium. TMR notices that the pre and post situations of two different recommendations are inverse of each other, therefore the one should be capable of mitigating an Adverse Drug Effect of the other.

+B4: Replacing a drug with a safer or non-interacting or more effective drug In case 1, TMR offers Clopidogrel as an alternative to Aspirin, which does not interact with any other recommendation or drug, making it a safer alternative. It should be noted that although the safer alternative is given, TMR does not automatically mark it as such in comparison with Aspirin. It is up to the user to make that distinction.

-B5: Discard unsafe/interacting drug This feature is not implemented. Similarly to the previous point, TMR creates the ability for the user to see alternatives to existing recommendations.

-C1: Patient preferences These features are not implemented. TMR focusses on the clinical aspects of guidelines, leaving patient preferences outside of the model's scope.

±C3: Explanation of the mitigation strategy(ies) This feature is partially implemented. Although TMR's output gives enough information to explain the mitigation strategies, the current version of TMR does not use this to create easy-to-read explanations.

+C4: Alternative mitigation strategies for a single interaction. C4 can be observed throughout the whole of case 1 above. There are alternatives found for most interactions. More specifically, TMR looks at which transitions are caused by a specific drug and compares it to other drugs, by querying the Drugbank API. If any drugs cause the same transitions then they are given as alternatives.

5 Comparison

As shown in the table below (which combines the scores across all three cases), TMR supports 10 out of the 18 functional features, 2 of which are supported partially. The unsupported features are [A6, A7, B2, B5, B6, B7, C1, C2]. The other models all support at least 10 features , which partially overlap with TMR.

The main differences are in features [A6, A7, B6, B7] which all deal with the temporal aspect of clinical guidelines and feature [B2] which deals with drug dosages. TMR is capable of supporting features [A5, B3, C3, C4] which does make it stand out from two of the six models.

Of particular interest is the comparison with SDA [5], since SDA is based on state-transitions to model medical guidelines. SDA scores considerably higher on many of the functional features. The strength of TMR is the fact that in SDA every interaction must be explicitly specified by the author of the SDA model, while TMR is able to automatically detect such interactions.

	A1	A2	A3	A4	A5	A6	A7	B1	B2	B3	B4	B5	B6	B7	C1	C2	C3	C4
TMR	+	+	+	+	+	-	-	+	-	±	+	-	-	-	-	-	±	+
SDA	+	+	+	+	+	+	+	+	+	+	+	+	+	+	±	-	+	+
GLARE-SCPM	+	+	+	±	+	+	+	+	+	+	+	±	+	+	+	-	+	+
PROforma-CMM	+	+	+	+	-	+	+	+	+	+	+	-	+	+	+	-	+	+
Cocom	+	+	+	+	+	+	+	+	+	-	+	+	+	+	+	-	+	+
Mitplan	+	+	±	+	±	+	+	+	+	+	+	+	+	+	+	±	-	±
CigIntO	+	+	±	+	+	+	+	+	+	+	+	+	+	±	-	+	-	-

We also compare TMR against the other guideline models in terms of the evaluation dimensions from [7], summarised in the table below. Of the six other methods, only two fully cover a category of a particular evaluation dimension. TMR is part of that as it fully covers the category for automated detection of adverse interactions between clinical guidelines. This makes it one of two methods to fully support it, while half of the methods could not even partially support this dimension. Shared with five other methods, TMR also has a logic-based foundation. In the case of TMR this is first-order logic.

	DET		STRAT			IMPL			HUM			
	complex reasoning	querying sources	adaptation	predefined algorithms	CIG language modules	planning	graph-based	logic-based	explanations	mitigation support	pull-based	push-based
TMR	+	+	-	+	-	-	-	+	+	-	+	-
SDA	-	-	-	-	+	-	+	+	+	+	+	-
GLARE-SSCPM	+	-	-	+	-	+	+	+	+	+	+	+
PROforma-CMM	-	-	-	-	+	-	-	+	+	+	+	-
GoCom	+	+	-	+	-	-	+	-	+	+	+	-
MitPlan	-	-	+	-	-	+	-	+	-	-	+	-
CigIntO	-	+	+	-	-	+	-	+	-	-	-	+

6 Conclusion

Evaluating TMR. Our evaluation has shown that the TMR model for computer interpretable clinical guidelines is only one of two methods able to automatically detect adverse interactions (the first evaluation dimension), using generic, reusable predefined first-order logic rules. However, it falls short on the functional features that require the representation of time and drug dosage. Furthermore, TMR computes a list of interactions, but stops short of providing recommendations on how to resolve these interactions. For future work, TMR should also be extended with temporal aspects and drug dosage. This should be possible given that both the state descriptions and the interaction rules are expressed in first-order logic.

Evaluating the Evaluation Methodology. Our study is the first independent validation of the evaluation methodology published in [7]. We have found that this methodology is executable in practice, and it enables the evaluation and comparison of the methods based purely on their functionality, independent of the specific implementation. Furthermore, the four cases of multi-guideline interaction have been well chosen and together cover all functional features. A possible weakness is that 8 of the 18 functional features are only covered by a single case. More redundancy among the cases would make the evaluation more robust. For future work, we encourage others to apply this methodology to other computer interpretable guideline models, as well as providing additional cases to use during the evaluation.

References

1. Kogan, A., Peleg, M., Tu, S.W., Allon, R., Khaitov, N., Hochberg, I.: Towards a goal-oriented methodology for clinical-guideline-based management recommendations for patients with multimorbidity: GoCom and its preliminary evaluation. J. Biomed. Informatics **112**, 103587 (2020)
2. Lozano, E., Marcos, M., Martínez-Salvador, B., Alonso, A., Alonso, J.R.: Experiences in the development of electronic care plans for the management of comorbidities. In: Riaño, D., ten Teije, A., Miksch, S., Peleg, M. (eds.) KR4HC 2009. LNCS (LNAI), vol. 5943, pp. 113–123. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-11808-1_10
3. Michalowski, M., Rao, M., Wilk, S., Michalowski, W., Carrier, M.: MitPlan 2.0: enhanced support for multi-morbid patient management using planning. In: Tucker, A., Henriques Abreu, P., Cardoso, J., Pereira Rodrigues, P., Riaño, D. (eds.) AIME 2021. LNCS (LNAI), vol. 12721, pp. 276–286. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-77211-6_31
4. Piovesan, L., Terenziani, P., Molino, G.: GLARE-SSCPM: an intelligent system to support the treatment of comorbid patients. IEEE Intell. Syst. **33**(6), 37–46 (2018)
5. Riaño, D.: The SDA model: a set theory approach. In: 20th IEEE International Symposium on Computer-Based Medical Systems (CBMS), pp. 563–568 (2007)
6. Woensel, W.V., Abidi, S.S.R., Abidi, S.R.: Decision support for comorbid conditions via execution-time integration of clinical guidelines using transaction-based semantics and temporal planning. Artif. Intell. Med. **118**, 102127 (2021)
7. Woensel, W.V., et al.: A community-of-practice-based evaluation methodology for knowledge intensive computational methods and its application to multimorbidity decision support. J. Biomed. Inform. **142**, 104395 (2023)

8. Zamborlini, V., Hoekstra, R., da Silveira, M., Pruski, C., ten Teije, A., van Harmelen, F.: A conceptual model for detecting interactions among medical recommendations in clinical guidelines. In: Janowicz, K., Schlobach, S., Lambrix, P., Hyvönen, E. (eds.) EKAW 2014. LNCS (LNAI), vol. 8876, pp. 591–606. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-13704-9_44
9. Zamborlini, V., Hoekstra, R., Silveira, M.D., Pruski, C., ten Teije, A., van Harmelen, F.: Inferring recommendation interactions in clinical guidelines. Semant. Web **7**(4), 421–446 (2016)

Frequent Patterns of Childhood Overweight from Longitudinal Data on Parental and Early-Life of Infants Health

Beatriz López[1](✉), David Galera[1], Abel López-Bermejo[2], and Judit Bassols[2]

[1] EXiT Research Group, University of Girona, Girona, Spain
beatriz.lopez@udg.edu

[2] Girona Biomedical Research Institute Dr. Josep Trueta (IDIBGI), Girona, Spain
{alopezbermejo,jbassols}@idibgi.org

Abstract. Childhood obesity is considered one of the main public health concerns. Research in the field of obesity detection and prevention is moving towards promising solutions thanks to the use of Artificial Intelligence applied to data from cohorts of children. Previous studies have analyzed the data without taking into account the relationship of data regarding when they are collected. In this work, frequent pattern mining is used to find the risk factors of childhood obesity, taking into account the relationship among the data gathered in different visits. The experiments carried out on the data collected from 386 children from Girona and Figueres (Spain) demonstrate the relevance of discriminant frequent patterns for childhood overweight prediction.

Keywords: Childhood obesity · Frequent pattern mining · Discriminant patterns

1 Introduction

Childhood obesity is one of the most important public health problems of the 21st century worldwide due to its prevalence and its impact on both short and long-term health [6]. Artificial Intelligence, and in particular Machine Learning approaches, have been used to understand the factors of childhood obesity [1]. Most of the approaches conduct cross-sectional studies in which socioeconomic and healthcare data of the parents and children are analyzed to understand the key factors of obesity. Few approaches dealt with longitudinal data. One of the exceptions is [4] that uses prenatal, perinatal, postnatal and 2 to 90 months age infant data, to identify different groups of risks. Nevertheless, [4] does not consider the relation of the variables according to the different visits in which they have been gathered. In our research we employ frequent pattern mining approaches, so that sets of variables are identified as key features instead of

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 64–69, 2024.
https://doi.org/10.1007/978-3-031-66538-7_7

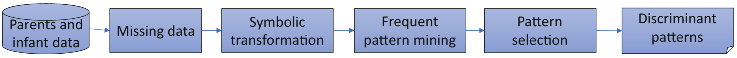


Fig. 1. Overview of the methodology.

variables in isolation. Moreover, managing longitudinal data poses a challenge in dealing with missing values, because the follow-up of the visits is sometimes discontinued. On the other hand, feature selection methods are required to achieve a good discriminating representation of patterns regarding the target variable (overweight/obesity or normal weight). So the contribution of this work involves coping with all the challenges to apply frequent pattern mining to the longitudinal data for characterizing childhood overweight.

2 Related Work

The work [4] used longitudinal data similar to the one proposed in this work, and [3] utilized electronic health record (EHR) data from the first two years of life to predict the obesity status at age five. However, none of these works take into account characterizing the baby according to frequent patterns as we do. A more recent approach is [8], which accounts for a similar imbalanced dataset as ours (14% of obese children), and reported the results using the AUC value as a performance metric. Conversely, we are analysing recall together with precision and accuracy, taking into account the minority class.

The use of frequent pattern mining is not new in Medicine, however. For example, [2] uses pattern mining algorithms to identify trajectories of patients from EHR.

3 Materials and Method

The methodology we propose is shown in Fig. 1. The dataset contains two cohorts of babies, from Hospital Dr. Josep Trueta (Girona) and Hospital of Figueres, both in Spain, collected from 2008 to 2014 (study with code 2010056 of the Clinical Research Ethics Committee of the Girona University Hospital Dr. Josep Trueta). Data were provided for 1175 infants, 212 variables each. Doctors who have been part of the project, select 36 variables as being the most related to the study, including variables related to the moment the mother gets pregnant, the information regarding the pregnant trimesters, the information when the mother give birth, and when the baby has 2 months until 6 years old. The target variable is OBSESITY, which indicates if the baby has a normal weight (OBSESITY=1) or has overweight or obesity (OBSESITY=2), according to the BMI measured when the child is 5 years old.

Missing Data Processing. First, the samples of the infants where values were missing in more than 60% of the variables were eliminated. Next, given the dataset D, for each numerical variable X_i that contains missing values, a subset of variables S, $X_i \in S$, with a high correlation with X_i has been identified. Second, a subset $D_i \subset D$ is obtained with all the instances that have the variable X_i also without information (missing). Third, for each instance I_j of D_i, a subset A is selected with the variables of S, $A \subseteq S$, that for the sample I_j do not have any missing value, $A \cap \{NA\} = \emptyset$. Fourth, a subset $D_j \subset D$ is gathered from the original dataset D, in which all the instances do not have missing values either in X_i or A, $D_j \cap \{NA\} = \emptyset$. Finally, a regression model is built from the dataset D_j to predict X_i. This process is repeated for each Instance I_j and missing value X_i. Regarding categorical data, we follow a similar process as for numerical variables, but using the highest correlation variable instead of a regression model.

Symbolic Transformation. Variables and their values should be transformed into symbols in order to apply pattern mining methods. Numerical variables (a total of 25), have been discretized each one in 4 bins. Boolean variables (8) have been transformed into one symbol per value, and the remainder categorical variables (2), one symbol per category.

Frequent Pattern Mining. Frequent pattern algorithms are applied to each class separately. The DefMe algorithm [5] has been used, with a support threshold of 0.3. From the two sets of frequent patterns, the symmetrical difference of the 2 sets is calculated. Finally, the original dataset is transformed using a binary variable for each pattern. For each instance, the value of a binary variable is either 1 or 0, depending on whether this pattern is present or not.

Pattern Selection. First we obtain a ranking of the patterns by using the variant of the mRMR algorithm described in [7], and next we follow an incremental process in which the different patterns are successively considered, according to their ranking, to build a predictive model. The performance of the model is visualized and with the collaboration of the medical team, the best N patterns are finally selected. An example of frequent pattern is the following: [$56 \leq$ Height at 2 months < 58, Smoking father $= 0$, 3 kg $\leq$ Birth weight mother $\leq$ 4 kg].

4 Results

To test our methodology, a total of three experiments have been carried out: Baseline, Sensitive analysis and Hybrid versus frequent patterns alone.

Baseline. We compare the results obtained with the plain dataset and the results obtained using the top 20 ranking frequent patterns. A 5-fold cross-validation is used 386 times (equivalent to the number of instances). Figure 2 shows the histograms obtained. The results are conclusive, the model makes better predictions if it is trained with the most discriminant patterns.

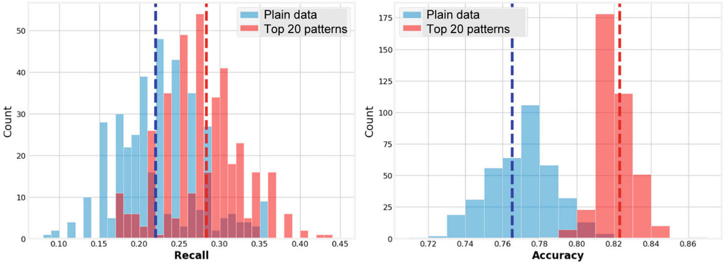


Fig. 2. Results: Value added by patterns.

Sensitive Analysis. We analyse the contribution of the addition of frequent patterns in the dataset (Fig. 3 (purple line)). Analyzing the recall of this second experiment, it can be deduced that using the first 20 patterns, the capacity of the model to detect instances of the obese (minority) class increases progressively, which confirms the discriminatory power of the first 20 patterns.

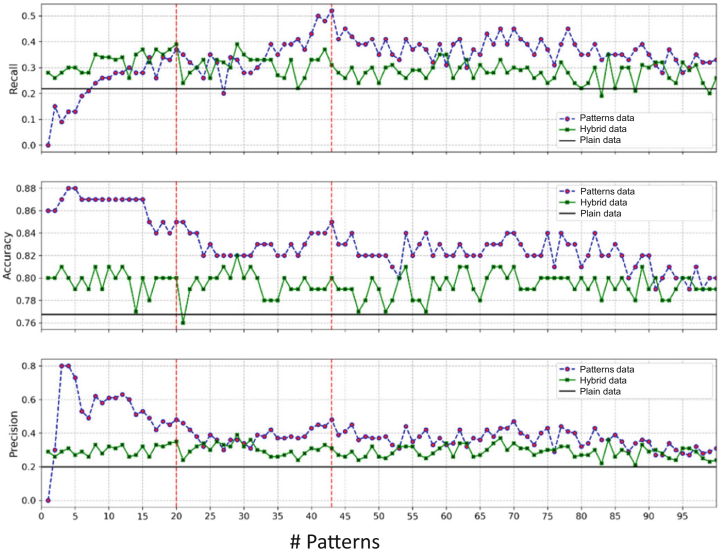


Fig. 3. Results: Sequential pattern selection (Color figure online).

Hybrid Versus Frequent Patterns Alone. We consider combining plain data with patterns in a hybrid dataset (Fig. 3). All the values obtained in the experiment are higher than expected using only the plain dataset.

Discussion. Adding frequent patterns considerably increases the quality of predictions compared to using only the original dataset. Regarding the results obtained for the accuracy measure (0.82), we can see that are close to the results of previous works: [8] 0.83 AUC; [3] 81.7 AUC for girls, and 76.1 for boys. On the other hand, we consider the recall measure due to the high imbalance dataset we have (14% of obese infants). The recall score is better if used in conjunction with the original data (hybrid) when using a quantity of patterns lower than the top 20 in the ranking.

Limitations. The results obtained represent certain limitations in clinical practice, due to the low values obtained in the performance metrics. One possible direction under study is to deal with gradient values of variables between visits, instead of actual ones, and use sequence pattern mining algorithms.

5 Conclusions

Childhood obesity is a disease that continues to grow around the world in an alarming way. This work proposes the use of frequent patterns learned from a dataset that contains longitudinal data of children and parents. The results of the experiments demonstrate the great discriminatory power of frequent patterns that capture the relationship between the data, versus other techniques. However, the results obtained have limited performance values. One of the lines of future work could be to explore alternative methods as sequence pattern mining algorithms.

Acknowledgements. We would like to thank Marina Rodriguez for her support in the initial dataset exploration. This work received joint funding from the European Regional Development Fund (ERDF), the Spanish Ministry of the Economy, Industry and Competitiveness (MINECO) and the Carlos III Research Institute, under grants no. PI23/00545 and PI22/00366. The work was carried out with support from the Generalitat de Catalunya 2021 SGR 01125.

References

1. Gou, H., Song, H., Tian, Z., Liu, Y.: Prediction models for children/adolescents with obesity/overweight: a systematic review and meta-analysis. Prev. Med. **179**, 107823 (2 2024). https://doi.org/10.1016/J.YPMED.2023.107823
2. Guillamet, G.H., Seguí, F.L., Vidal-Alaball, J., López, B.: Cauruler: Causal irredundant association rule miner for complex patient trajectory modelling. Comput. Biol. Med. **155**, 106636 (3 2023). https://doi.org/10.1016/J.COMPBIOMED.2023.106636

3. Hammond, R., Athanasiadou, R., et. al: Predicting childhood obesity using electronic health records and publicly available data. PLOS ONE **14**, e0215571 (4 2019). https://doi.org/10.1371/JOURNAL.PONE.0215571
4. O'Connor, T.G., Williams, J., et al.: Predictors of developmental patterns of obesity in young children. Front. Pediatr. **8**, 506691 (3 2020). https://doi.org/10.3389/FPED.2020.00109/BIBTEX
5. Soulet, A., Rioult, F.: Efficiently depth-first minimal pattern mining. LNAI **8443**, 28–39 (2014) https://doi.org/10.1007/978-3-319-06608-0_3/COVER
6. WHO: Obesity and overweight (2024). https://www.who.int/news-room/fact-sheets/detail/obesity-and-overweight
7. Zhao, Z., Anand, R., Wang, M.: Maximum relevance and minimum redundancy feature selection methods for a marketing machine learning platform. In: Proceedings of the IEEE International Conference on Data Science and Advanced Analytics (DSAA), pp. 442–452 (10 2019).https://doi.org/10.1109/DSAA.2019.00059
8. Ziauddeen, N., Roderick, P.J., Santorelli, G., Alwan, N.A.: Prediction of childhood overweight and obesity at age 10-11: findings from the studying lifecourse obesity predictors and the born in bradford cohorts. Int. J. Obesity 2023 47:11 **47**, 1065–1073 (8 2023). https://doi.org/10.1038/s41366-023-01356-8

Fuzzy Neural Network Model Based on Uni-Nullneuron in Extracting Knowledge About Risk Factors of Maternal Health

Paulo Vitor de Campos Souza(✉) and Mauro Dragoni

Fondazione Bruno Kessler, Trento, Italy
{pdecampossouza,dragoni}@fbk.eu

Abstract. The potential of diagnostic methodologies to improve pregnant women's well-being is critical in preventing complications for mothers and their unborn children. This paper introduces an interpretable fuzzy neural network model that uses artificial intelligence (AI) techniques for early risk detection in pregnancy. The model, which integrates a fuzzy inference system and a defuzzification process across three layers, provides deep insights by formulating fuzzy rules from the data. Comparative analysis with existing models shows that our approach achieves high accuracy in maternal risk identification and offers enhanced interpretability and detailed data analysis.

1 Introduction

Maternal health analysis is crucial for improving outcomes for expectant mothers and children. Utilizing artificial intelligence (AI), significant insights into risk factors are gained through data analytics, which helps in early detection and personalized interventions, aiming to reduce maternal mortality globally [2].

AI is pivotal in identifying maternal health risks via data collection technologies. Despite its promise, the lack of interpretability in some AI methods may cause uncertainty among users. Fuzzy neural networks, which merge fuzzy system clarity with neural network training simplicity, are notably effective in this domain [4].

The paper details a novel Interpretable Fuzzy Neural Network Uni-Nullneuron-Based Approach (IFNN-Uninull), tailored for classification and exploring dataset relationships. This model incorporates a fuzzification process, Gaussian neurons, and uni-null neurons to establish interpretable fuzzy if-then rules, thus enhancing model understandability [8]. Its efficacy in maternal health risk identification is validated against existing models [1].

2 Literature Review

AI and its Influence on Maternal Risk Assessment. Maternal health risks such as advanced maternal age and adolescent pregnancies are influenced

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 70–75, 2024.
https://doi.org/10.1007/978-3-031-66538-7_8

by socioeconomic and ethnic disparities, significantly affecting mortality rates. AI technologies enhance maternal healthcare by enabling remote monitoring and early detection of complications, thus improving access in underserved areas. Predictive models and AI-driven interventions like chatbots and telemedicine help address these disparities, promoting personalized and inclusive healthcare solutions [6].

Fuzzy Neural Networks Concepts. Fuzzy neural networks (FNNs) combine neural networks with fuzzy systems to improve decision-making in complex scenarios. These networks utilize fuzzification processes to handle data and generate fuzzy rules, useful in expert systems development. The structure of FNNs varies, with configurations ranging from three to five layers, allowing for customized solutions tailored to specific problems [10].

3 IFNN-Uninull Architecture and Training

This paper presents a new Fuzzy Neural Network (FNN) architecture that improves interpretability and feature alignment through innovative neuron designs and weight modifications [3]. The model consists of three layers: a fuzzification layer using Gaussian neurons to reflect data density, a middle layer with uni-nullneurons for efficient fuzzy inference, and a final neural network layer that synthesizes rules using leaky-ReLU functions. This enhanced FNN structure maintains all fuzzy rules for thorough knowledge extraction from the data.

3.1 First Layer: Data Density Fuzzification Approach

The IFNN-UniNull model's first layer employs an advanced fuzzification process using Gaussian neurons to convert inputs into fuzzy sets, thus improving flexibility [7]. This setup, including the Self-organizing Direction-Aware Data Partitioning (SODA) algorithm, adapts to changing data patterns effectively, using methods like cosine similarity and Euclidean distance for optimal data partitioning [5,7].

Inputs are transformed into membership degrees across L fuzzy sets per variable, enhancing the model's responsiveness to new data inputs. The output for each neuron is derived from a structured fuzzification formula:

$$\Omega(x_j, c_{jl}, \sigma_{jl}) = e^{-\frac{1}{2}\left(\frac{x_j - c_{jl}}{\sigma_{jl}}\right)^2}, \text{ for } j = 1...N,\ l = 1...L, \tag{1}$$

where N is the number of inputs, L the number of fuzzy sets per input, c_{jl} the center, and σ_{jl} the standard deviation for each Gaussian neuron.

3.2 Definition of Weights in the First Layer Neurons: Feature Weight Calculation

De Campos Souza and Lughofer [5] enhance fuzzy rule interpretability in their novel modeling approach by linking weight assignments directly to data features using the Dy-Brodley criterion [8] (to reduce dimensionality, effectively streamlining learning and rule development). This criterion highlights the feature importance for class differentiation, improving clarity and rule conciseness. Weights in the first layer are defined as:

$$w_{il},\ i = 1\dots N,\ l = 1\dots\ L \tag{2}$$

indicating class separability.

Separability is quantified by:

$$J = \delta(S_w^{-1}S_b), \tag{3}$$

reflecting the sum of diagonal components of $S_w^{-1}S_b$, where S_b and S_w denote the between-class and within-class scatter matrices, respectively.

3.3 Second Layer: Fuzzy Rules

The second layer of the IFNN-Uninull model employs uni-nullneurons for transforming data into IF-THEN rules using Type-III fuzzy logic. These neurons use a combination of t-conorms and t-norms in their operations, enabling flexible rule creation with AND and OR connectors:

$$N^{U-NU}(x,y) = \begin{cases} 0 & ,(x,y)\ \in [0,g[^2, \\ \max\ (x,y) & ,(x,y)\ \in [g,u]^2, \\ u & ,(x,y)\ \in [g,u] \times]u,1] \cup]u,1] \times [g,u] \cup]u,1[^2, \\ \min(x,y) & elsewhere. \end{cases} \tag{4}$$

Outputs from these neurons are aggregated into fuzzy rules that enhance interpretability and classification flexibility. Weights and activation levels from the first layer are integrated using a relevancy function to optimize rule precision and adaptability [5].

$$p(w,a,g,u) = (w \vee g) \wedge (\bar{w} \vee u) \wedge (a \vee u), \tag{5}$$

Rule synthesis is performed as follows, illustrating the modular and adaptable nature of rule construction:

$$\begin{aligned} Rule_L :\ If\ x_1\ is\ A_L^1\ with\ impact\ w_{1L}\dots \\ AND/OR_{(g,u)}\ x_2\ is\ A_L^2\ with\ impact\ w_{2L}\dots \\ Then\ y_L\ is\ [v_{L1}\dots v_{LC}] \end{aligned} \tag{6}$$

The defuzzification and output layer aggregation utilize leaky-ReLU functions to finalize the output based on the fuzzy rule results, ensuring dynamic response to input variations [9].

$$\hat{y} = \left(\sum_{j=0}^{L} f_{LeakyReLU}(z_j, v_j) \right), \tag{7}$$

reflecting the comprehensive approach to managing class-specific activations and rule effectiveness in the model.

4 Experiment

4.1 Data Set Characteristics

We utilized a diabetes patient dataset specifically curated for analyzing common risk factors and categorized them into risk levels with indicators such as Age, SystolicBP, DiastolicBP, BS, BodyTemp, and HeartRate [1].

4.2 Models and Test Premises

The experiment evaluated the IFNN-Uninull model's performance in pattern classification against traditional models (e.g., K-nearest neighbors, Naive Bayes) and other neuro-fuzzy models using a grid partition value of 4 and cross-validation with γ values ranging from 2 to 7. This setup was aimed at establishing the model's effectiveness in handling complex data structures, using overall accuracy as the evaluation metric. We used standard parameters provided by Orange software for all models in the comparison.

4.3 Test Results

Table 1 compares traditional and fuzzy models' performance; the best scores are in bold, asterisks indicate scores comparable to our model. Experiments used a Core 2 Duo CPU, 2.27 GHz, 3 GB RAM setup.

Table 1. Result of maternal health classification.

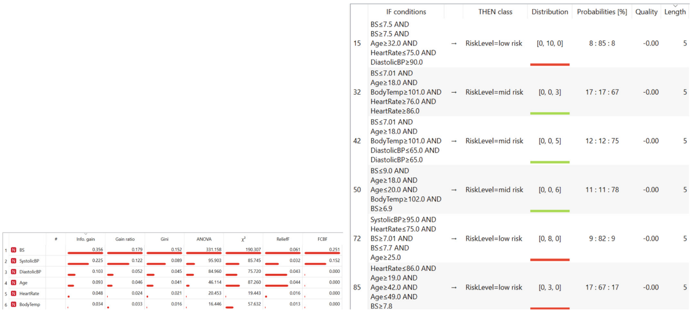
Model	IFNN-Uninull	RFNN	DDFNN
Accuracy (%)	79.33 (3.05)	73.96 (2.36)	38.47 (4.91)
Model	UNI null-FNN	SGD	KNN
Accuracy (%)	74.67 (3.19)	63.39 (1.14)	67.52 (0.03)
Model	TR	SVM	RFR
Accuracy (%)	76.78 (0.25)	47.11 (2.23)	79.17 (0.14)*
Model	NN	NB	LR
Accuracy (%)	67.71 (2.16)	66.62 (2.04)	62.34 (1.15)
Model	GB	CN2	
Accuracy (%)	78.70 (0.03)*	78.94 (0.02)*	

4.4 Discussions

Table 1 demonstrates that our proposed model excels in maternal risk identification compared to other neural networks and fuzzy models. This is evidenced by the development of 210 rules, an here is an example:

1. If (Age is small) 0.26 and (Systolic BP is medium) 0.68 and (Diastolic BP is small) 0.67 and (BS is high) 1.00 and (Body Temp is medium) 0.68 and (Heart Rate is medium) 0.67 then (Risk Level is low risk = **high possibility** and mid risk = **small possibility** and high risk = **small possibility**).

IFNN-Uninull uses 250 fuzzy rules with "and/or" connectors for clear multi-class classification in maternal risk. The SODA method prunes unnecessary clusters, and Gaussian functions ensure complete coverage. Gaussian weights highlight feature relevance, as shown in Fig. 1 (a).



		#	Info. gain	Gain ratio	Gini	ANOVA	χ²	ReliefF	FCBF
1	N BS		0.356	0.179	0.152	331.158	190.307	0.061	0.251
2	N SystolicBP		0.225	0.122	0.089	95.903	85.745	0.032	0.152
3	N DiastolicBP		0.103	0.052	0.045	84.960	75.720	0.043	0.000
4	N Age		0.093	0.046	0.041	46.114	87.260	0.044	0.000
5	N HeartRate		0.048	0.024	0.021	20.453	19.443	0.016	0.000
6	N BodyTemp		0.034	0.033	0.016	16.446	57.632	0.013	0.000

	IF conditions		THEN class	Distribution	Probabilities [%]	Quality	Length
15	BS≤7.5 AND BS≥7.5 AND Age≥32.0 AND HeartRate≤75.0 AND DiastolicBP≥90.0	→	RiskLevel=low risk	[0, 10, 0]	8 : 85 : 8	-0.00	5
32	BS≤7.01 AND Age≥18.0 AND BodyTemp≥101.0 AND HeartRate≥76.0 AND HeartRate≥86.0	→	RiskLevel=mid risk	[0, 0, 3]	17 : 17 : 67	-0.00	5
42	BS≤7.01 AND Age≥18.0 AND BodyTemp≥101.0 AND DiastolicBP≤65.0 AND DiastolicBP≥65.0	→	RiskLevel=mid risk	[0, 0, 5]	12 : 12 : 75	-0.00	5
50	BS≤9.0 AND Age≥18.0 AND Age≤20.0 AND BodyTemp≥102.0 AND BS≥6.9	→	RiskLevel=mid risk	[0, 0, 6]	11 : 11 : 78	-0.00	5
72	SystolicBP≥95.0 AND HeartRate≤75.0 AND BS≥7.01 AND BS≤7.7 AND Age≥25.0	→	RiskLevel=low risk	[0, 8, 0]	9 : 82 : 9	-0.00	5
85	HeartRate≤86.0 AND Age≥19.0 AND Age≥42.0 AND Age≤49.0 AND BS≥7.8	→	RiskLevel=low risk	[0, 3, 0]	17 : 67 : 17	-0.00	5

Fig. 1. Feature evaluation scores and Rules extracted by CN2 rule inducer.

The IFNN-Uninull model prioritizes crucial dimensions like Blood Sugar (BS) for assessing maternal risk, marking it with the highest weight, whereas age is shown to have a minimal impact. This focus enhances interpretability by simplifying rules and aiding in clearer problem understanding. For instance, the model captures that high BS combined with moderate Systolic BP, Diastolic BP, Body Temp, and Heart Rate typically indicates a low risk level, simplifying complex risk assessments into understandable terms.

In contrast, the CN2 inducer's use of numerical rules and strict AND connectives limits flexibility and interpretability compared to IFNN-Uninull's mixed AND/OR structure, as shown in Fig. 1 (b). This difference highlights IFNN-Uninull's superior capability to offer clear, linguistic explanations of risk factors, facilitating easier comprehension of intricate data relationships.

5 Conclusions and Future Work

This study improved transparency in maternal risk identification using fuzzy rules that clarify feature interrelationships and significance during pregnancy. The rules' certainty values and fuzzy sets offer insights into risk levels and feature importance, enhancing interpretability compared to traditional models. Future work will focus on refining training methods and weight adjustments to boost this model's clarity and effectiveness.

References

1. Ahmed, M., Kashem, M.A., Rahman, M., Khatun, S.: Review and analysis of risk factor of maternal health in remote area using the internet of things (IoT). In: Kasruddin Nasir, A.N., Ahmad, M.A., Najib, M.S., Abdul Wahab, Y., Othman, N.A., Abd Ghani, N.M., Irawan, A., Khatun, S., Raja Ismail, R.M.T., Saari, M.M., Daud, M.R., Mohd Faudzi, A.A. (eds.) InECCE2019: Proceedings of the 5th International Conference on Electrical, Control & Computer Engineering, Kuantan, Pahang, Malaysia, 29th July 2019, pp. 357–365. Springer Singapore, Singapore (2020). https://doi.org/10.1007/978-981-15-2317-5_30
2. Bosschieter, T.M., et al.: Unique insights into risk factors for antepartum stillbirth using explainable AI. Am. J. Obst. Gynecol. **228**(1), S403–S404 (2023)
3. de Campos Souza, P.V., Guimaraes Nunes, C.F., Guimares, A.J., Silva Rezende, T., Araujo, V.S., Silva Arajuo, V.J.: Self-organized direction aware for regularized fuzzy neural networks. Evol. Syst. **12**(2), 303–317 (2021)
4. de Campos Souza, P.V.: Fuzzy neural networks and neuro-fuzzy networks: a review the main techniques and applications used in the literature. Appl. Soft Comput. **92**, 106275 (2020)
5. de Campos Souza, P.V., Lughofer, E.: An evolving neuro-fuzzy system based on uni-nullneurons with advanced interpretability capabilities. Neurocomputing **451**, 231–251 (2021)
6. Feduniw, S., et al.: Application of artificial intelligence in screening for adverse perinatal outcomes-a systematic review. In: Healthcare. vol. 10, p. 2164. MDPI (2022)
7. Gu, X., Angelov, P., Kangin, D., Principe, J.: Self-organised direction aware data partitioning algorithm. Inf. Sci. **423**, 80–95 (2018)
8. Lughofer, E.: On-line incremental feature weighting in evolving fuzzy classifiers. Fuzzy Sets Syst. **163**(1), 1–23 (2011), theme: Classification and Modelling
9. Maas, A.L., Hannun, A.Y., Ng, A.Y.: Rectifier nonlinearities improve neural network acoustic models. In: Proc icml. vol. 30, p. 3 (2013)
10. Pedrycz, W., Gomide, F.: Fuzzy systems engineering: toward human-centric computing. John Wiley & Sons (2007)

Identifying Factors Associated with COVID-19 All-Cause 90-Day Readmission: Machine Learning Approaches

Shiwei Lin[1,2,3], Shiqiang Tao[1,2], Yan Huang[1,2], Xiaojin Li[1,2](✉),
and Guo-Qiang Zhang[1,2](✉)

[1] Department of Neurology, McGovern Medical School, The University of Texas Health Science Center at Houston, Houston, TX 77030, USA
{xiaojin.li,guo-qiang.zhang}@uth.tmc.edu

[2] Texas Institute for Restorative Neurotechnologies, The University of Texas Health Science Center at Houston, Houston, TX 77030, USA

[3] Department of Biostatistics and Data Science, School of Public Health, The University of Texas Health Science Center at Houston, Houston, TX 77030, USA

Abstract. The COVID-19 pandemic has placed immense strain on healthcare systems. In response to this challenge, our study employs machine learning techniques to identify and analyze risk factors associated with COVID-19 all-cause 90-day readmission. Leveraging the Optum® de-identified COVID-19 Electronic Health Record data set, we developed predictive models with comparable efficacy, particularly the optimized XGBoost model in prognosticating readmission risks. Our analysis reveals several key risk factors aligned with existing research and finds specific laboratory tests that may serve as potential indicators of readmission risk. By elucidating these critical determinants, our study expands the knowledge base for clinical decision-making, offering healthcare practitioners deeper insights into the factors affecting COVID-19 patient readmission risks. These findings can potentially empower clinicians to refine interventions and care strategies, mitigating adverse outcomes and advancing healthcare delivery for individuals affected by COVID-19.

Keywords: COVID-19 · Readmission Prediction · Machine Learning

1 Introduction

The coronavirus disease 2019 (COVID-19) has resulted in over six million hospitalizations and one million deaths as of February 2024 in the United States [4]. These overwhelming incidents pressured healthcare systems tremendously, increasing medical supply and resource demands [2]. Hospital readmission, a long-standing and costly concern in public health, has become more critical in the context of the pandemic [9]. As a fundamental indicator of healthcare quality, managing COVID-19-related readmission is paramount to conserving medical resources and ensuring patient safety. Several studies

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 76–80, 2024.
https://doi.org/10.1007/978-3-031-66538-7_9

[6, 11, 12] have explored the characteristics of the readmitted COVID-19 patient population and investigated related risk factors for readmission. While sample sizes vary, prevalent risk factors such as congestive heart failure (CHF), chronic kidney disease (CKD), chronic obstructive pulmonary disease (COPD), and diabetes emerge consistently across studies. However, these studies are constrained by limited sample sizes of readmitted patients and a narrow scope of risk factors, with relatively few risk factors linked to laboratory tests. Although the 30-day readmission rate is a widely used quality metric for hospital care, most readmissions for COPD patients occur after this period [5]. Moreover, studies on 90-day readmission are limited and the risk factors influencing 90-day readmission may differ. These limitations emphasize the necessity for comprehensive research using large-scale datasets with more readmitted patients included in the analysis.

In this paper, we utilize the extensive Optum® de-identified COVID-19 Electronic Health Record data set to explore and identify risk factors associated with COVID-19 90-day all-cause readmission, employing advanced machine learning techniques. Our study discovered key risk factors are consistent with established literature, validating our methodology. Furthermore, our analysis uncovered additional potential risk factors contributing to a deeper understanding of COVID-19 90-day readmission dynamics. This research offers valuable insights into the factors associated with COVID-19 readmission and the potential of machine learning in improving prognostic accuracy and guiding clinical decision-making.

2 Methods

A positive COVID-19 case is confirmed by a patient with either: (1) a diagnosis with ICD-10 code U07.1, or (2) a positive result from a COVID-19 related Polymerase Chain Reaction (PCR) test, antibody test, or antigen test, or (3) a manually reviewed positive lab test. Admission is determined for a COVID-19 patient's visit exceeding 24 hours, with either a corresponding COVID-19 diagnosis or a positive PCR test result. All-cause 90-day readmission is defined as inpatient readmission occurring 1 to 90 days after initial admission discharge. Readmissions occurring within 24 hours of discharge or lasting less than 24 hours are excluded. In case multiple readmissions are observed, only the one with the earliest visit start date is considered. To efficiently process and retrieve the Optum® COVID-19 data, we apply the Event-level Inverted Index [8], a fast temporal query method. The initial extraction yields 1,498,140 patients between February 1, 2020, and January 20, 2022. Only admitted patients with matching COVID-19 diagnoses or positive PCR tests are retained for this study. Exclusions are applied to patients lacking related visit records and whose visit duration is under 24 hours. Following these criteria, 46,106 COVID-19 patients with valid admission records remain for further analysis. The primary outcome of this study is all-cause 90-day readmission status, "Yes" for readmitted patients and "No" for not readmitted patients.

A total of 33 features are included in the analysis. The demographic features contain age group, gender, race, and ethnicity. Six diagnosis features identified by existing research [6, 11, 12] are included: CHF, CKD, COPD, diabetes, pneumonia, and sepsis. Ten laboratory test results are comprised in the analysis: alanine transaminase (ALT), aspartate aminotransferase (AST), blood urea nitrogen, creatinine, estimated glomerular rate (eGFR), hematocrit (HCT), hemoglobin (HGB), platelet count (PLT), red blood cell count (RBC), and white blood cell count (WBC). The inclusion of ALT, AST, creatinine, eGFR, and PLT is based on their potential to predict adverse outcomes in COVID-19 patients, as indicated by existing research [1, 3, 13, 14]. Three observational features are entailed: body mass index (BMI), respiratory rate (RESP), and systolic blood pressure (SBP). We also incorporate ten aggregated features to quantify each patient's medical history of visits, diagnoses, and laboratory tests [10]: (1) length of stay (LOS) in days; (2) total diagnoses one year before the initial admission; (3) total unique diagnoses one year before the initial admission; (4) total diagnoses during the initial admission (DIAG_AD); (5) total unique diagnoses during the initial admission (DIAG_ UNIQ_ AD); (6) total visits one year before the initial admission (VIS_1YR); (7) total lab tests one year before the initial admission (LAB_1YR); (8) total lab tests with results outside the normal range one year before the initial admission; (9) total lab tests during the initial admission (LAB_AD); and (10) total lab tests with results outside the normal range during the initial admission (LAB_AD_N).

Patients lacking the results of all laboratory tests and all observational records are eliminated from the study to ensure data completeness. A mean value imputation is implemented to address the remaining missing data. After data preprocessing, the study consists of 43,610 patients, with 17,659 (40.49%) experiencing readmission, while 26,041(59.71%) patients were not readmitted.

We develop k-nearest neighbors (kNN), support vector machine (SVM), random forest (RF), and XGBoost (eXtreme Gradient Boosting) models to predict COVID-19 patient readmission. The feature importance derived from both RF and XGBoost models provides critical insights into the determinants of readmission risk. All the models are trained using the same set of 33 features. Bayesian optimization is applied to find the optimal hyper-parameters for the XGBoost and RF models. Model performance is assessed using the area under the receiver operating characteristic curve (AUROC) and prediction accuracy.

3 Results

Table 1 displays the performance of six models trained to analyze risk factors for COVID-19 90-day all-cause readmission. The optimized XGBoost model outperformed, achieving a mean accuracy of 0.72 with an AUROC of 0.75 over ten runs. Figure 1 presents the key features identified by the optimally tuned XGBoost and RF models, with seven features consistent between them. Laboratory tests are prominent among the key features. The optimized XGBoost model highlights seven laboratory-related features, and the optimized RF model reveals eight important laboratory-related features.

4 Discussion and Conclusion

The feature importance displayed in Fig. 1 demonstrates the consistency of our findings with previous research, reinforcing the validity of our predictive models. Notably: (1) we identify chronic kidney disease as a risk factor, aligning with the study by Verna et al.

Table 1. Performance of the trained models based on ten runs.

Model	kNN	SVM	XGBoost (default)	XGBoost (optimized)	RF (default)	RF (optimized)
Accuracy	0.58	0.60	0.68	**0.72**	0.66	0.68
AUROC	0.60	0.62	0.70	**0.75**	0.68	0.70

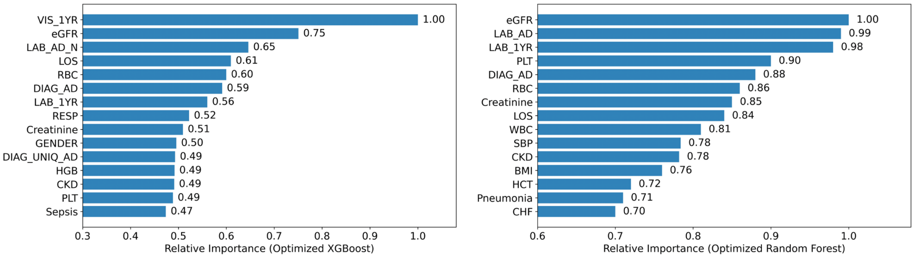


Fig. 1. Feature importance of the optimized XGBoost and random forest models.

[12], which links CKD to higher readmission odds; (2) the eGFR is recognized as a key laboratory test feature supporting the findings of Appelman et al. [1], which indicates a direct correlation between lower eGFR levels and higher readmission rates; (3) our analysis of length of stay data suggests that shorter hospitalization duration is associated with an increased risk of readmission, which is in line with findings reported by Weaver et al. [13]; (4) our investigation of serum creatinine levels as a risk factor is consistent with the results of Yeo et al. [14]. Their study reveals that patients with high serum creatinine levels during initial hospital stay had a higher risk of 30-day readmission; (5) we also detect an association between PLT and increased risks of 90-day readmission, consistent with findings by Boccatonda et al. [3], which links decreased PLT to increased critical illness and all-cause mortality in hospitalized COVID-19 patients.

Our analysis suggests that RBC and WBC could be prognostic factors. Previous study [7] has highlighted the correlation between elevated WBC counts at admission and increased mortality rates in COVID-19 patients, suggesting that higher WBC levels should be monitored and managed during COVID-19 treatment. Furthermore, the aggregated features employed in our study have revealed that patient history can assist in predicting readmission risks. For instance, the count of past hospital visits emerges as the most influential feature in the optimized XGBoost model. For future work, we could incorporate more features to enhance the models' comprehensiveness and predictive accuracy, such as more diagnoses related to COVID-19 and patients' prescriptions. Moreover, employing deep learning models could allow the exploration of complex nonlinear relationships and the extraction of intricate features.

Acknowledgments. This work was supported in part by the National Institutes of Health (NIH) grant R01NS126690. The content is solely the responsibility of the authors and does not necessarily represent the official views of the NIH.

Disclosure of Interests. The authors have no competing interests to declare relevant to this article's content.

References

1. Appelman, B., Oppelaar, J.J., Broeders, L., Wiersinga, W.J., Peters-Sengers, H., Vogt, L.: Mortality and readmission rates among hospitalized COVID-19 patients with varying stages of chronic kidney disease: a multicenter retrospective cohort. Sci. Rep. **12**(1), 1–8 (2022)
2. Birkmeyer, J.D., Barnato, A., Birkmeyer, N., Bessler, R., Skinner, J.: The impact of the COVID-19 pandemic on hospital admissions in the United States. Health Aff. **39**(11), 2010–2017 (2020)
3. Boccatonda, A., et al.: Platelet count in patients with SARS-CoV-2 infection: a prognostic factor in COVID-19. J. Clin. Med. Res. **11**(14) (2022)
4. CDC: COVID data tracker, March 2020. https://covid.cdc.gov/covid-data-tracker/
5. Dong, F., Huang, K., Ren, X., Wang, Y., Jiao, J., Yang, T.: Factors associated with 90-day readmission in inpatients with chronic obstructive pulmonary disease. Eur. Respir. J. **56**(Suppl. 64) (2020)
6. Guarin, G., et al.: Factors associated with hospital readmissions among patients with COVID-19: a single-center experience. J. Med. Virol. **93**(9), 5582–5587 (2021)
7. Haji Aghajani, M., et al.: Risk factors of readmission in COVID-19 patients; a retrospective 6-month cohort study. Arch. Acad. Emerg. Med. **10**(1), e48 (2022)
8. Huang, Y., Li, X., Zhang, G.Q.: ELII: a novel inverted index for fast temporal query, with application to a large covid-19 EHR dataset. J. Biomed. Inform. **117**, 103744 (2021)
9. Kangovi, S., Grande, D.: Hospital readmissions—not just a measure of quality. JAMA **306**(16), 1796–1797 (2011)
10. Kim, Y., et al.: Characterizing cancer and COVID-19 outcomes using electronic health records. PLoS ONE **17**(5), e0267584 (2022)
11. Menditto, V.G., et al.: Predictors of readmission requiring hospitalization after discharge from emergency departments in patients with COVID-19. Am. J. Emerg. Med. **46**, 146–149 (2021)
12. Verna, E.C., et al.: Factors associated with readmission in the United States following hospitalization with coronavirus disease 2019. Clin. Infect. Dis.. Infect. Dis. **74**(10), 1713–1721 (2021)
13. Weaver, F.M., et al.: Hospital readmissions among veterans within 90 days of discharge following initial hospitalization for COVID-19. Prev. Chronic Dis. **19**, E80 (2022)
14. Yeo, I., et al.: Assessment of thirty-day readmission rate, timing, causes and predictors after hospitalization with COVID-19. J. Intern. Med. **290**(1), 157–165 (2021)

Mining Disease Progression Patterns for Advanced Disease Surveillance

Syed Hamail Hussain Zaidi(✉), Amna Basharat, and Muddassar Farooq

CureMD Research, 80 Pine St 21st Floor, New York, NY 10005, USA
hamail1041@gmail.com, amna.basharat@nu.edu.pk, muddassar.farooq@curemd.com
https://www.curemd.com/

Abstract. In this paper, we present a smart disease surveillance system that reveals insights into disease trajectories indicating risks of subsequent diseases starting from the current health conditions, for individual patients or populations. Using a pattern mining algorithm, we extract disease trajectory patterns from temporally modeled encounters of 17 million patients in the medical knowledge graph and develop a disease surveillance system on 477,933 mined patterns of disease progression. The system predicts future disease trajectory of individual patients and facilitates in-depth exploration into disease mechanisms, root causes and future disease progression at a patient cohort level thereby enabling early interventions for complex diseases and promoting an evidence based precision medicine approach for healthcare providers.

Keywords: Disease Trajectories · Knowledge Graph · Pattern mining · Data Science

1 Introduction

The exponential growth in the availability of medical data, ranging from structured EMR/EHRs to unstructured patient notes, has opened new horizons in the field of medicine. The ongoing revolution in precision medicine, characterized by treatment and preventive measures tailored to the health profile of individual patients, is made possible by advancements in data analytics and machine learning. One catalyst enabling this revolution is the capability of an AI model to predict patient's disease trajectories accurately [1] from the current health profile of a patient. It empowers healthcare providers to extract new insights from big healthcare data comprising of electronic health records.

Population level analytics of patient journeys and disease progressions have been carried out in the past by focusing only a few diseases and limited applications [4]. A statistical study [3] extracted 1,171 trajectories consisting of a strong temporal pair of any two diseases to find disease progression. Earlier studies have also used network approaches that use specific demographics i.e. for people older than 65 over a selected study period (i.e. 2–3 years) [2]. To the best of our knowledge, little efforts are devoted to building disease and/or demographics agnostic systems to provide a holistic overview of disease trajectories of patients.

Supported by CureMD.

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 81–89, 2024.
https://doi.org/10.1007/978-3-031-66538-7_10

Our research introduces a disease-agnostic surveillance system that not only predicts future disease trajectories for individual patients based on past and current diagnoses, but also uncovers comprehensive disease trajectory patterns across patient groups.

2 Methodology

The proposed disease surveillance system, shown in Fig. 1, presents an end-to-end framework designed to harness temporally modelled patient encounters within medical knowledge graphs. The key stages of the disease trajectories pipeline are described in the following sections.

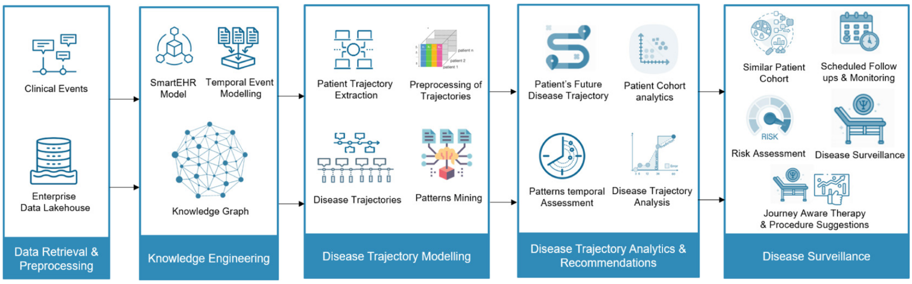

Clinical Events
Enterprise Data Lakehouse
Data Retrieval & Preprocessing
SmartEHR Model
Temporal Event Modelling
Knowledge Graph
Knowledge Engineering
Patient Trajectory Extraction
Preprocessing of Trajectories
Disease Trajectories
Patterns Mining
Disease Trajectory Modelling
Patient's Future Disease Trajectory
Patient Cohort analytics
Patterns temporal Assessment
Disease Trajectory Analysis
Disease Trajectory Analytics & Recommendations
Similar Patient Cohort
Scheduled Follow ups & Monitoring
Risk Assessment
Disease Surveillance
Journey Aware Therapy & Procedure Suggestions
Disease Surveillance

Fig. 1. High Level Architecture of Disease Surveillance system.

Knowledge Engineering. The system begins by constructing a temporally modelled knowledge graph from clinical events stored in an Enterprise Data Lakehouse of CureMD. The knowledge graph contains 17 million patients and all of their encounters attached to them temporally. Nodes in the graph represent clinical entities, such as encounters, diagnoses (ICD codes), prescriptions, and vitals etc. Edges represent relationships between these entities. A sample patient's temporally modelled clinical encounters in knowledge graph is shown in Appendix A (Fig. 5). It represents the temporal progression, with specific timestamps indicating when a particular set of clinical events occurred for a given patient. This allows us to track the evolution of a patient's health status.

Disease Trajectories Retrieval and Pre-processing. Using the medical knowledge graph, we extract a patient's disease trajectory from temporally modelled encounters. For every encounter, we fetch 2 things: (1) ICD codes (disease diagnosed), and (2) Date of the encounter in which different procedures were performed by physicians. From these 17 million trajectories, we then shortlist a set of patients having at least 3 encounters. The number of shortlisted patients matching the above-mentioned criteria is approximately 3 million. A sample disease progression trajectory is shown in Appendix B.

Temporal Pattern Mining. After extracting and pre-processing disease trajectories, we use a pattern mining algorithm (shown in Appendix C) to mine disease progression patterns of varying lengths from 3 million patient's disease trajectories. These trajectories represent sequences of clinical encounters for individual patients and the algorithm extracts prevalent patterns from these sequences It also keeps track of the patients' demographics while mining the patterns to ensure in-depth analytics enabling future analyses and retrospection studies for each mined pattern.

Pattern Analysis. After mining all the patterns of lengths 3,4 and 5, all the patient identifiers attached to each of them are back tracked for their demographics analysis to filter any bias in the patterns. Age and gender distribution of patients are analysed and stored with each pattern. So now, for each pattern (e.g. R53.8 → N40.1 → E11.9), we have the following information: (1) *support,* is the number of patients who experienced the same disease progression pattern;(2) *time distribution* is the time to disease progression for each patient; (3) *age distribution* of patients supporting a given disease progression pattern; and (4) *gender distribution* of the patients supporting a given disease progression pattern.

3 Results

A total of 477,933 prevalent patterns, supported by 10+ patients each, were mined from 247 million unique disease patterns of 3 million patients. Some example patterns are tabulated in the Table 1. We take latest 3 encounters of the patient, match the so far disease trajectory of the patient in the patterns and then extrapolate the trajectory based on the mined patterns.

Table 1. Overview of Mined Patterns

Pattern	Progression	Time (days)	Patient Support	Avg. Age	Gender Distribution
K21.9 → K29.70 → B96.81	K21.9 → K29.70 K29.70 → B96.81	3 weeks 2 weeks	547	55y	M: 47% F: 53%
K64.8 → K21.9 → K44.9	K64.8 → K21.9 K21.9 → K44.9	1 year 3 weeks	437	66y	M: 41% F: 59%
E78.5 → I73.9 → I87.2 → R09.89	E78.5 → I73.9 I73.9 → I87.2 I87.2 → R09.89	2 weeks 1 week 2 months	107	73y	M: 34% F: 66%
I10 → I73.9 → I87.2 → R09.89	I10 → I73.9 I73.9 → I87.2 I87.2 → R09.89	1 month 4 weeks 6 months	122	74y	M: 34% F: 66%

3.1 Disease Surveillance for Individual Patients

Using patients' past and current medical conditions, the system identifies patterns and correlations that may indicate the likelihood of subsequent health

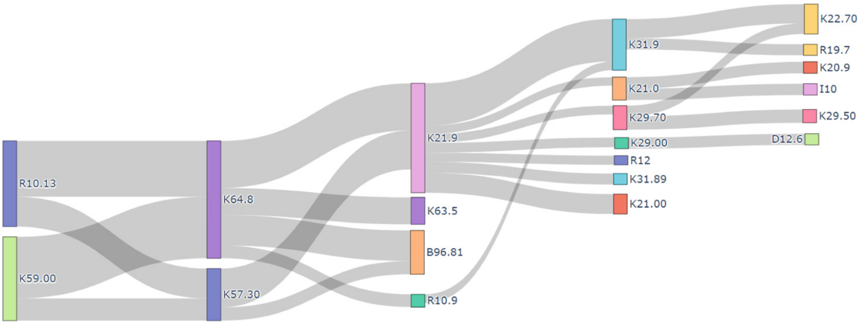


Fig. 2. Sample Patient's future Disease Trajectory

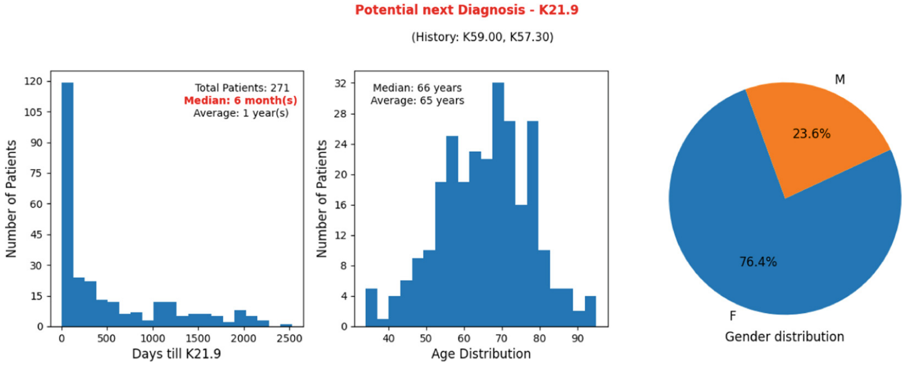


Fig. 3. Pattern analysis of next potential diagnosis - K21.9

conditions. Lets consider a patient who is suffering from a functional agastrointestinal disorder (K59.00), unspecified abdominal pain (R10.13), other hemorrhoids (K64.8), and diverticular disease of the intestine (K57.30). Figure 2 shows in detail the future disease trajectory for this patient based on current medical conditions. The nodes represent the ICD codes, and the thickness of each pathway is indicative of the number of patients who have followed that specific trajectory. Figure 3 provides a time distribution analysis indicating how quickly patients might progress to the next diagnoses. The median time to a 'K21.9' diagnosis is 6 months (Fig. 3), whereas for 'B96.81', it is 4 weeks (shown in Appendix D, Fig. 6), highlighting the rapid progression of certain conditions in patients.

3.2 Patient Cohort Analysis

The disease surveillance system also facilitates cohort-level analysis for understanding the progression and root causes of medical conditions across a large patient population of interest.

Trajectories to Colon Cancer: Mapping Prevalent Precursors. This insight is of value for clinicians to explore the root causes of a particular disease, say Malignant neoplasm of cancer (C18.9). Our system identifies most prevalent antecedent conditions to colon cancer which will aid clinicians to proactively intervene. Prevalent disease trajectories leading to a colon cancer are shown in Fig. 4. It involves conditions like adenomatous polyps of the colon (D12.5), intra-abdominal and pelvic swelling (R19.09), Polyp of colon (K63.5) etc. suggesting that persistent gastrointestinal issues or certain benign polyps could potentially escalate to malignancy over time.

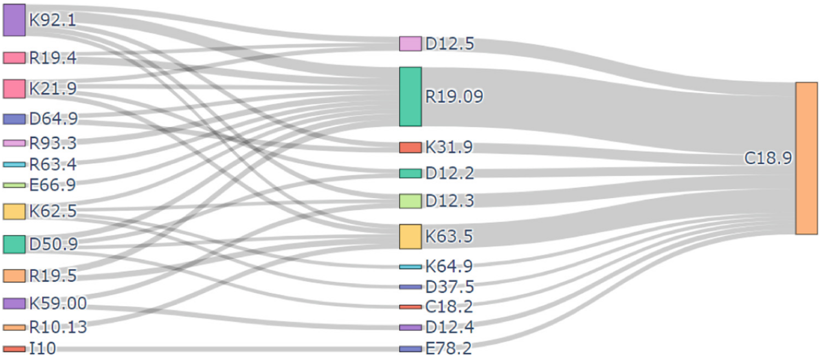


Fig. 4. Cohort level analysis of disease trajectories leading to Colon Cancer

3.3 Discussion

In addressing the challenges associated with evaluating our system, we implemented a retrospective analysis to establish the probabilistic support for each identified trajectory in addition to the absolute number of patients following that pathway. This dual-parameter approach ensures that our predictions are not only based on the pathways supported by a relatively large number of patients but are also statistically significant. Last but not least, a panel of three expert physicians from a partner healthcare facility has clinically validated all the use cases presented in this paper.

4 Conclusion and Future Work

We introduced a disease surveillance system for predicting disease progression trajectories by temporally modelling the clinical encounters of 17 million patients in a knowledge graph. A novel algorithm identifies patterns in disease progression, enabling personalized prediction tools for healthcare professionals. The system includes patient specific disease trajectories as well as cohort level disease progression trends pinpointing prevalent precursors and aftereffects of a particular set of diseases, thus allowing early interventions. Future work will expand the disease surveillance system to modelling the efficient and effective treatment pathways.

A Temporally Modelled Patient's Clinical Data in KG

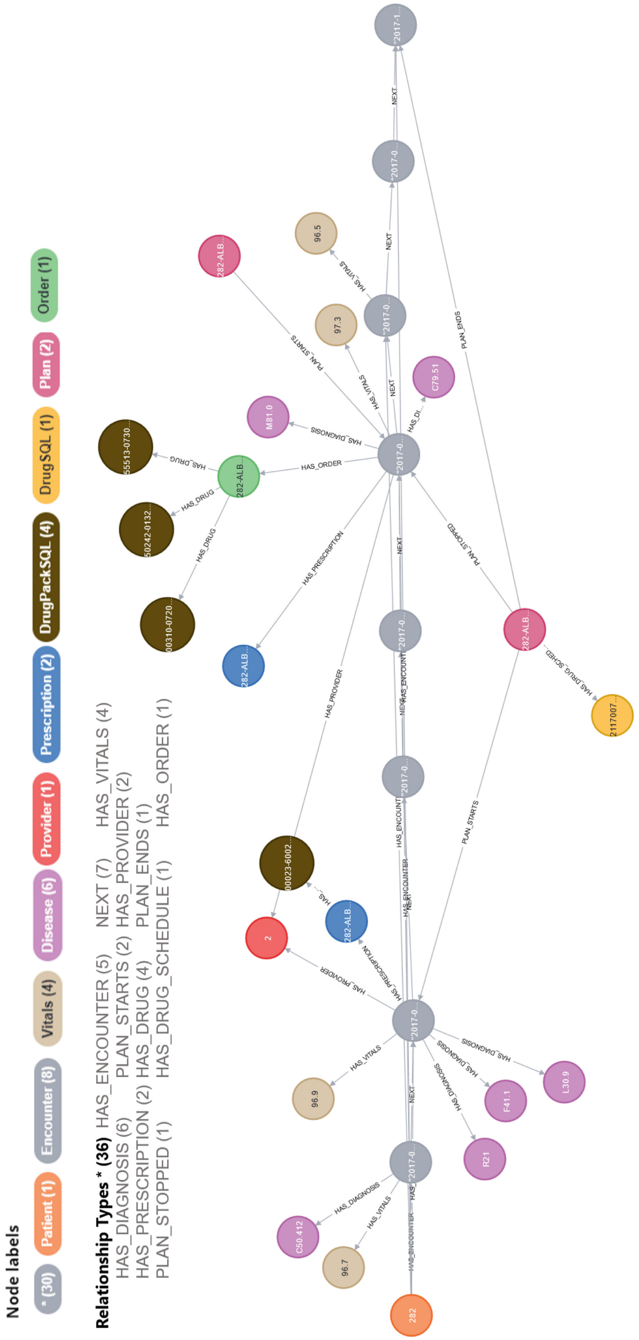


Fig. 5. Temporal modelling of Patient's clinical data using Graph Based approach

B Sample Disease Trajectory from KG

See (Table 2)

Table 2. A sample disease trajectory of a patient from the Knowledge Graph

Encounter #	Date	Diagnosis	Description
1	2019-12-26	E03.9 E66.01	Hypothyroidism, unspecified Morbid (severe) obesity due to excess calories
2	2020-01-13	E55.9	Vitamin D deficiency, unspecified
3	2020-02-26	E78.5 I10 E53.8	Hyperlipidemia, unspecified Essential (primary) hypertension Deficiency of other specified B group vitamins
4	2020-04-04	E11.9	Type 2 diabetes mellitus without complications

C Algorithm for Disease Patterns Mining

Algorithm 1. Mine Encounter-Wise Disease Patterns

```
Require: A list of patient trajectories, $\mathcal{T} = \{T_1, T_2, ..., T_n\}$, and granularity $\mathcal{G}$.
Ensure: Sorted mapping of encounter-wise patient cohort patterns, $\mathcal{P}_{sorted}$.
Initialize $\mathcal{P}$ as an empty mapping.
for each trajectory $T_i \in \mathcal{T}$ do
    Initialize a set $S_i$ for patient pattern.
    Extract encounter dates $D_i = \{d_1, d_2, ..., d_m\}$ from $T_i$,
    Calculate day differences $\Delta D_i = \{\delta_1, \delta_2, ..., \delta_{m-1}\}$, where $\delta_j = d_{j+1} - d_j$.
    for each combination $C$ in the Cartesian product of ICD codes from encounters in $T_i$ do
        for each $n$-gram $\gamma$ from $\Gamma_{\mathcal{G}}(C)$ do
            if $|\text{unique}(\gamma)| = \mathcal{G}$ then
                Append $(\gamma, \Delta D_i)$ to $S_i$.
            end if
        end for
    end for
    for each pattern $p = (\gamma, \Delta D_i)$ in $S_i$ do
        if $\gamma$ exists in $\mathcal{P}$ then
            Append $\Delta D_i$ to $\mathcal{P}[\gamma]$.
        else
            Set $\mathcal{P}[\gamma] = [\Delta D_i]$.
        end if
    end for
end for
return $\mathcal{P}_{sorted}$, the sorted version of $\mathcal{P}$.
```

D Patient Level Analytics

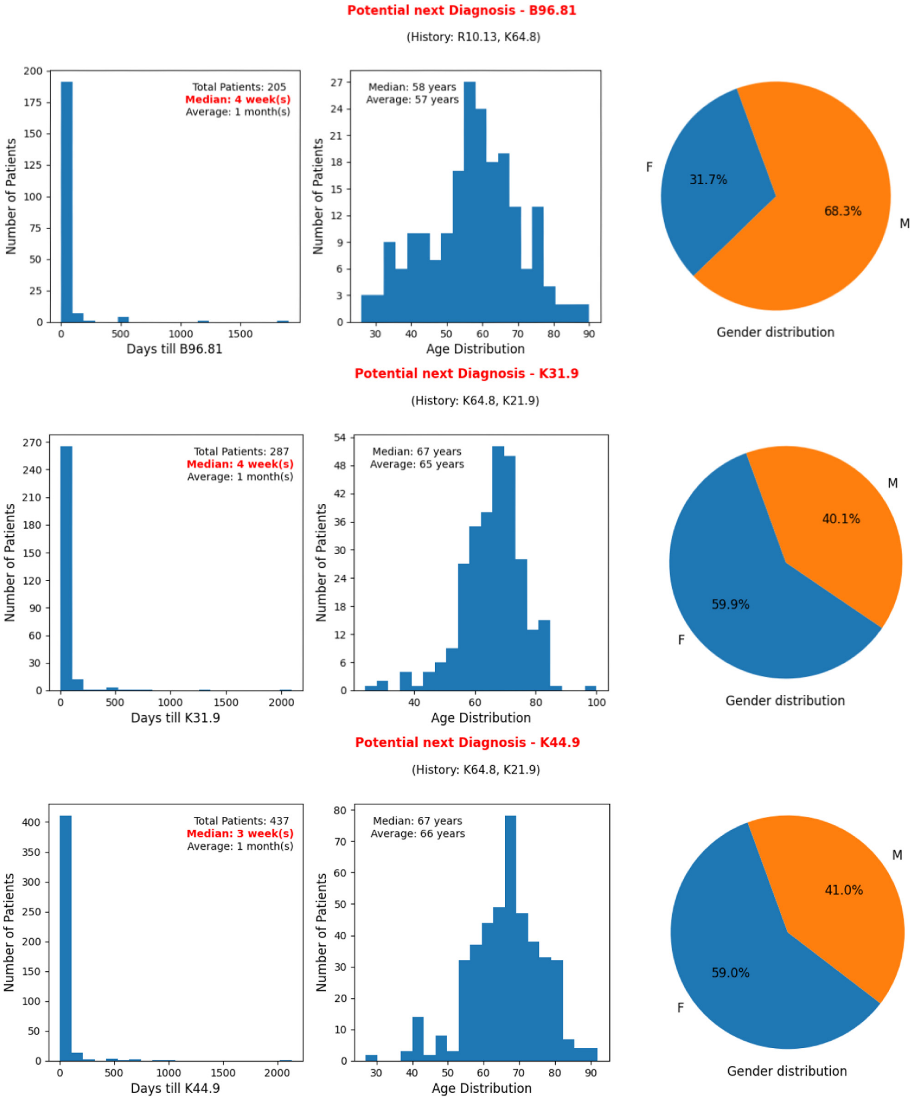


Fig. 6. Pattern analysis of next potential diagnoses for sample patient

References

1. Aspland, E., Gartner, D., Harper, P.: Clinical pathway modelling: a literature review. Health Syst. **10**(1), 1–23 (2021)
2. Hidalgo, C.A., Blumm, N., Barabási, A.L., Christakis, N.A.: A dynamic network approach for the study of human phenotypes. PLoS Comput. Biol. **5**(4), e1000353 (2009)
3. Jensen, A.B., et al.: Temporal disease trajectories condensed from population-wide registry data covering 6.2 million patients. Nature Commun. **5**(1), 4022 (2014)
4. Murtagh, F.E., Murphy, E., Sheerin, N.S.: Illness trajectories: an important concept in the management of kidney failure (2008)

Minimizing Survey Questions for PTSD Prediction Following Acute Trauma

Ben Kurzion[1], Chia-Hao Shih[2], Hong Xie[2], Xin Wang[2], and Kevin S. Xu[1](✉)

[1] Case Western Reserve University, Cleveland 44106, USA
{bxk389,ksx2}@case.edu
[2] University of Toledo, Toledo 43614, USA
{chiahao.shih,hong.xie,xin.wang2}@utoledo.edu

Abstract. Traumatic experiences have the potential to give rise to post-traumatic stress disorder (PTSD), a debilitating psychiatric condition associated with impairments in both social and occupational functioning. There has been great interest in utilizing machine learning approaches to predict the development of PTSD in trauma patients from clinician assessment or survey-based psychological assessments. However, these assessments require a large number of questions, which is time consuming and not easy to administer. In this paper, we aim to predict PTSD development of patients 3 months post-trauma from multiple survey-based assessments taken within 2 weeks post-trauma. Our objective is to *minimize the number of survey questions* that patients need to answer while *maintaining the prediction accuracy from the full surveys*. We formulate this as a feature selection problem and consider 4 different feature selection approaches. We demonstrate that it is possible to achieve up to 72% accuracy for predicting the 3-month PTSD diagnosis from 10 survey questions using a mean decrease in impurity-based feature selector followed by a gradient boosting classifier.

Keywords: Survey optimization · PTSD prognosis · Feature selection · Mean decrease in impurity · Gradient boosting · Random forest

1 Introduction

Post-traumatic stress disorder (PTSD) is a complex somatic, cognitive, affective and behavioral disorder that emerges in response to traumatic life events. PTSD is characterized by intrusive thoughts, nightmares and flashbacks of past traumatic events, avoidance of reminders of trauma, hypervigilance, and sleep disturbance, all of which lead to considerable social, occupational, and interpersonal dysfunction [2].

Diagnosis of PTSD requires patients to undergo expensive and time consuming clinical tests with specialists. On the other hand, traditional survey-based psychological assessments are relative inexpensive and easy to administer. We

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 90–100, 2024.
https://doi.org/10.1007/978-3-031-66538-7_11

consider the possibility of training a machine learning (ML) algorithm to predict the PTSD diagnosis from the clinician using the survey data rather than having patients frequently go through structured clinical interviews. However, these surveys can become very long and repetitive for patients and lead to fatigue.

In this study, we design a shortened survey that enables an ML algorithm to predict the PTSD diagnosis, ideally with a similar level of accuracy as using a full survey. We formulate this survey minimization task as a feature selection problem, where patients' responses to different survey questions are treated as the features. We explore four different approaches for feature selection: stability selection using the LASSO, two variants of mean decrease in impurity (MDI)-based selection, and a maximum depth limitation approach.

We find that an MDI-based feature selector followed by a gradient boosting classifier is able to predict the PTSD diagnosis of patients 3 months post-trauma with up to 72% accuracy using 10 questions from surveys taken within 2 weeks post-trauma. Importantly, we find that this level of accuracy is comparable to or even higher than the accuracy we obtain when training ML algorithms on the full survey data, indicating that we can *minimize the survey to 10 questions without losing prediction accuracy.*

2 Background

2.1 Feature Selection for Machine Learning Algorithms

The objective of a supervised ML algorithm is to accurately predict a target y given a feature vector $\boldsymbol{x}$ representing an object. Given a set of training examples $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n\}$ and associated targets $\{y_1, y_2, \ldots, y_n\}$, an ML algorithm learns a function f such that, for a new test example $\boldsymbol{x}^*$, $f(\boldsymbol{x}^*)$ is a good predictor of its target y^*. In the context of this paper, the object would be a patient, the target would be the patient's PTSD diagnosis, and the feature vector denotes relevant information about the patient, such as the patient's age, sex, and responses to survey questions. Each subject is diagnosed as either PTSD or non-PTSD; in this setting, the ML algorithm is a binary *classifier*, and the target is a *label.*

It is often the case that the entire feature vector $\boldsymbol{x}$ may not be necessary to learn a good predictor $f(\boldsymbol{x})$. For example, some of the features may be irrelevant for predicting the label y. *Feature selection* algorithms aim to select a subset of features to use to train the ML model. They are used to discard irrelevant features from a classifier and result in a smaller model. They may also result in higher classification accuracy in some instances.

A variety of feature selection algorithms are often employed for training ML models. Some are generic "wrapper" algorithms that can be applied to any ML model, including greedy algorithms such as forward and backward subset selection [1]. Others have been designed specifically for a class of ML models. For example, the LASSO [24] is designed specifically for feature selection in linear predictors, such as linear and logistic regression. It is sometimes used to perform feature selection when using other non-linear machine learning algorithms. A related feature selection approach is stability selection, where a feature selector

Table 1. Demographic and trauma-related information on the 144 participants.

Information	Mean (SD) or Number (%)
Age	33.2 (10.8)
Sex (Female)	98 (68.1%)
Trauma type	
Motor vehicle accident	79 (54.9%)
Physical assault	53 (36.8%)
Sexual assault	8 (5.6%)
Other	4 (2.8%)
PTSD at 3 months post-trauma	54 (37.5%)

such as LASSO is repeatedly applied to randomly sampled subsets of data [18]. The features that are selected most frequently in the random subsets are then chosen as the features used to train the ML model.

2.2 Related Work

The related work most relevant to this paper focuses on survey optimization. In a survey optimization study, researchers employed a LASSO regression model to best approximate the five most effective questions to put in their survey about women's agency. They trained 1,000 LASSO regressions, each on a random 50% of the data, and tracked which features were selected by each model. Then, they chose the 5 most frequently selected features (i.e., survey questions) to use in their optimized survey [13], which is a form of stability selection. Their study aimed to find exactly 5 features, but it is possible to extend their ideas to find the optimal balance between relevant features and survey accuracy, which is our focus. A similar approach was employed for household poverty classification in [14], resulting in a survey with 10 questions. Finally, another related approach for survey optimization formulates the problem as a Markov Decision Process where possible actions are questions we can ask, states are the known answers, and the reward function is how well the current state can predict the outcome [15].

3 Data Description

3.1 Participants

We consider data collected from a longitudinal neuroimaging study [12,25] on adult trauma survivors who seek medical treatment. Participants completed initial evaluation regarding behavioral and clinical symptoms using questionnaires within 2 weeks post-trauma and underwent a clinical interview for PTSD diagnosis (CAPS-5) at 3 months post-trauma. The study was approved by the University of Toledo Institutional Review Board protocol #201575. All participants

Table 2. Survey assessments used as features in this paper. Participants responded to each survey question on a scale with the listed number of choices or scale points.

Survey	Questions	Scale pts
Acute stress disorder scale (ASDS) [5]	19	5
Quick Inventory of Depressive Symptomatology (QIDS) [21]	16	4
Pain Anxiety Symptoms Scale – Short Form (PASS) [17]	20	6
Difficulties in Emotion Regulation Scale (DERS) [11]	36	5
Childhood Trauma Questionnaire (CTQ) [3]	28	5
Pittsburgh Sleep Quality Index Addendum (PSQI-A) [10]	7	5

provided written informed consent and received monetary compensation for their participation in the study. Summary statistics about the participants are shown in Table 1.

3.2 Features

Based on the PTSD literature and previous findings using the data from the same cohort [7,8,22,23,26], we compile a total of 129 features. The features include participants' demographic and trauma-related information (i.e., age, sex, and trauma type), and behavioral and clinical symptoms in the form of responses to individual questions from 6 validated survey instruments, listed in Table 2.

3.3 Imputation of Missing Data

Roughly 2% of our data are missing, corresponding to survey questions that participants did not answer. The features with the highest rates of missing data are QIDS questions 6 and 8, which focus on decreased appetite and weight, respectively. They are followed by questions 7 and 9 on increased appetite and weight, which may have led participants to ignore questions 6 and 8. Question 6 had 6.25% missing data (3.47% from PTSD and 2.78% from non-PTSD participants), while question 6 had 5.56% missing data (3.47% PTSD, 2.08% non-PTSD).

In order to properly train and test our ML models, we perform data imputation for the missing data. We use the `IterativeImputer` from the scikit-learn Python package [20] to impute missing values on the entire data set. The imputation approach is a single regression-based multivariate imputation based on the Multivariate Imputation by Chained Equations (MICE) package in R [6].

4 Methodology

We first train a variety of classifiers on the full survey data to determine the attainable PTSD prediction accuracy using all 129 features. We then compare feature selection approaches to choose a smaller subset of features to use when training the classifiers and investigate the prediction accuracy using different feature selection methods and different numbers of features.

4.1 Classification on Full Survey Data

We assess these models using a 5 × 5 nested cross validation (CV), with 5 outer folds and 5 inner folds. The outer folds are used to evaluate the model's prediction accuracy, while the inner folds are used to select the hyperparameters used to train the models via grid search. The hyperparameter values that maximize the mean inner CV are chosen. These hyperparameters are then used to evaluate prediction accuracy in the outer CV. Both the outer and inner CV are stratified by the labels: the PTSD status at 3 months post-trauma. Using a nested CV rather than a single CV prevents us from using the same CV to choose hyperparameters and evaluate accuracy, which may result in overly optimistic prediction accuracy values that may not be achievable in practice.

We consider three different classification algorithms: logistic regression with an elastic net penalty [27], random forest [4], and gradient boosting [9]. Logistic regression makes a prediction using a linear function of the features, while random forest and gradient boosting are non-linear ensemble predictors that use multiple decision trees to arrive at a prediction. Logistic regression is sensitive to feature scaling, while random forest and gradient boosting are not, so we standardize the features for logistic regression only. The values and ranges we consider for key hyperparameters are shown in Table 3. Random forest and gradient boosting were trained using the Gini impurity and the mean squared error improvement criterion (`'friedman_mse'` in scikit-learn) [9], respectively, to measure the quality of a split.

Table 3. Hyperparameters and their ranges of values

Model	Hyperparameter	Values
Logistic Regression	l_1-ratio	$\{0.75, 0.85, 0.95\}$
	Inverse regularization strength C	$\{10^{-3}, 10^{-2}, \ldots, 10^{2}\}$
Random Forest	# of trees	$\{5, 10, \ldots, 100\}$
	Max. depth	$\{1, 2, \ldots, 9\}$
	Max. features	{None, `'sqrt'`, 0.2, 0.5, 0.7}
Gradient Boosting	# of trees	$\{5, 10, \ldots, 100\}$
	Max. depth	$\{1, 2, 3, 4\}$

4.2 Minimizing Survey Questions Using Feature Selection

We formulate the objective of minimizing the number of survey questions as a *feature selection* problem. Feature selection denotes selecting a subset of the available features to use when training a classifier. It is used to discard irrelevant features from a classifier and results in a smaller model. It may also result in higher classification accuracy, but we are more interested in the smaller model.

Any feature that gets discarded from the classifier corresponds to a survey question that *does not need to be asked.* Thus, if a feature selection approach results in only 5 questions selected, then those 5 questions now form the minimized survey and are the only questions that the subject would be presented with.

We experiment with a variety of feature selection methods, described below, to observe the relationship between classification accuracy and the number of features used. We treat the participant's age, sex, and trauma type as "always available" features since they are not survey questions. These 3 features are excluded from the feature selection process.

The LASSO stability selection and mean decrease in impurity (MDI) approaches are applied before training the classification model, while the maximum depth limitation approach modifies how the classifier is trained. For LASSO stability selection and MDI, we evaluate with the same 5×5 stratified, nested CV as we use for the models on the full survey data. In each inner fold, we train the given selector on the inner CV training set, rank the features, and then train the classifier (with a grid search for hyperparameter values) on the top k features for $k \in \{3, 5, 10, 13, 15, 20\}$. The maximum depth limitation approach does not have any additional hyperparameters and thus uses only a single CV, which is the same as the outer CV for the models on the full survey data.

LASSO Stability Selection: We train 1,000 logistic regression models with an ℓ_1 (LASSO) penalty on a random 50% of the inner fold's training data. We perform stability selection [18] by ranking each feature by the number of times it was selected by all 1,000 models. In order to rank the features, we tuned the logistic regression `C` hyperparameter such that the logistic regression model had non-zero coefficients for a large proportion of the total features (roughly over 30%).

Mean Decrease in Impurity (MDI): Tree-based models offer a "built-in" approach for measuring feature importance, the mean decrease in impurity (MDI) [16]. When the impurity measure used is Gini impurity, it also goes by the name Gini importance [19]. We use the MDI purely for feature selection by training a tree-based ensemble model on the inner fold training data and then ranking features by their importance in terms of the MDI. We consider two tree-based ensemble models for MDI-based feature selection: random forest and gradient boosting. In scikit-learn, the MDI-based is given by the attribute `feature_importances_`.

Note that we are training a random forest or gradient boosting model on the full survey data just for the purpose of feature selection. Similar to LASSO stability selection, we tune the number of trees used for random forest and gradient boosting such that they produce a non-zero importance for a large proportion of the total features (roughly over 30%). We discard the random forest or gradient boosting model and then train the actual classifier on only the selected features to evaluate prediction accuracy on the reduced survey.

Maximum Depth Limitation: This approach is designed for boosted decision trees, which typically use very shallow trees as the base classifier. If we constrain the maximum depth of the tree to be 1, turning it into a decision stump, then a

Table 4. Comparison of different measures of prediction accuracy and number of features used across different classifiers trained using all of the survey questions. Reported results are over a 5 × 5 nested cross-validation with mean ± standard error over the outer folds. Bold entries denote highest accuracy, sensitivity, specificity, and lowest number of features used.

Classifier	Accuracy	Sensitivity	Specificity	Features Used
Logistic Regression	0.653 ± 0.034	0.407 ± 0.061	0.800 ± 0.067	**37.4 ± 10.4**
Random Forest	0.659 ± 0.025	**0.513 ± 0.086**	0.744 ± 0.038	70.6 ± 15.0
Gradient Boosting	**0.666 ± 0.027**	0.402 ± 0.092	**0.822 ± 0.021**	38.0 ± 18.3

boosting model with L trees can make at most L splits, and hence, use at most L features. To use this maximum depth limitation with gradient boosting, we constrain the model to use maximum depth of 1 and vary the number of trees L according to Table 3. For each value of L, we record the number of features used. Since this method does not require any hyperparameter tuning, we use a single 5-fold cross validation to determine the accuracy for all values of L. Unlike the other feature selection approaches, this approach does not require first training an additional classifier in order to perform feature selection. It is thus much more computationally efficient than the other two approaches.

5 Results

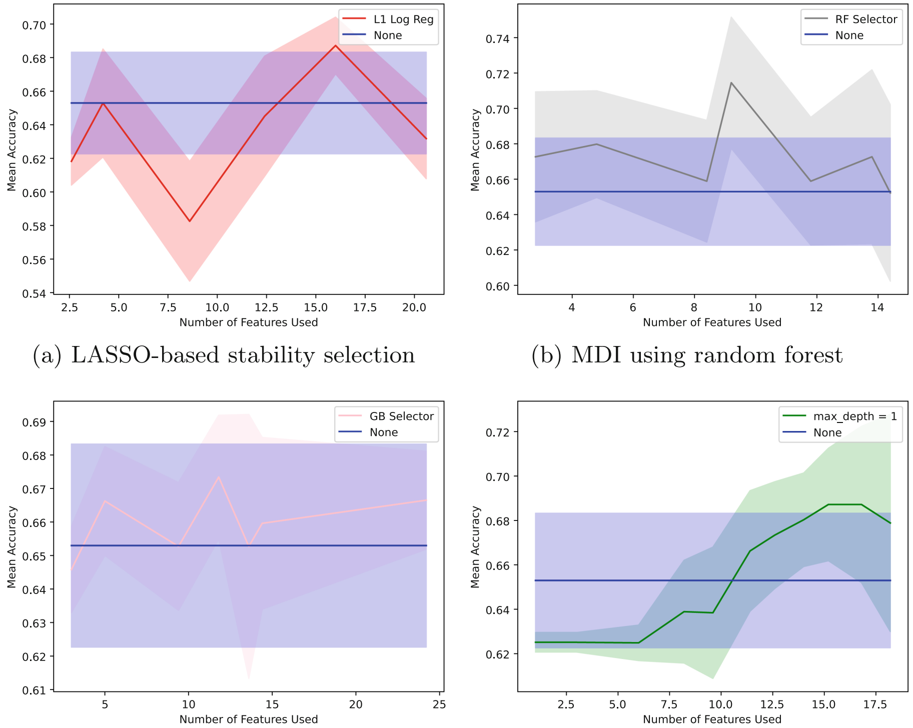
5.1 Prediction Using Full Survey Data

The prediction accuracy for the three different classifiers is shown in Table 4. All of the classifiers perform similarly in their prediction accuracy values, which are within 1 standard error of each other. All classifiers had higher specificity than sensitivity, likely due to the higher number of non-PTSD patients in the data.

While all of the models perform similarly in accuracy, we also consider how many features a model uses to inform its prediction. Since we are aiming to minimize the number of features by feature selection, choosing a classifier that already uses a smaller number of features makes for a good starting point. The model with the highest accuracy and lowest number of features is the best. Given these criteria, we choose gradient boosting as the classifier to use for survey minimization, as it has the highest accuracy and finishes a close second in the number of features used.

5.2 Prediction Using Shortened Surveys

We plot the mean prediction accuracy for varying numbers of features used in Fig. 1 for each of the four feature selection techniques. The mean accuracy values for all feature selectors are actually *higher* for some number of features than the corresponding values for gradient boosting on the full survey. For the MDI-based feature selection using random forest, increased accuracy is achieved



(a) LASSO-based stability selection

(b) MDI using random forest

(c) MDI using gradient boosting

(d) Maximum depth limitation

Fig. 1. Comparison of gradient boosting accuracy with no feature selection (blue) and using four different feature selection methods (other colors). Confidence bands denote 1 standard error for the mean accuracy over a 5-fold cross validation. (Color figure online)

using as few as 3 features, while the other approaches require more features. All are able to achieve higher accuracy than gradient boosting on the full survey once the number of features is in the 10-20 range. The highest values of prediction accuracy are shown in Table 5 for models with less than 5, 10, and 20 features.

These results suggest that it is indeed possible to *significantly reduce the length of the survey* while maintaining or even possibly improving the prediction accuracy. We note that, in most cases, the confidence band (±1 standard error) overlaps with the confidence band for gradient boosting without feature selection. This suggests that, while we cannot confirm that we are improving prediction accuracy, it is likely that the prediction accuracy using the shortened surveys is at least comparable to that using the full surveys. The feature selector with the highest mean accuracy is the MDI-based selector using random forest.

Table 5. Comparison of gradient boosting prediction accuracy using limited numbers of features for different feature selection methods. Bold entries denote highest accuracy for a fixed maximum number of features.

Feature Selector	≤5 Features	≤10 Features	≤20 Features
LASSO stability selection	0.653 ± 0.0322	0.653 ± 0.0322	0.687 ± 0.017
MDI using random forest	**0.680 ± 0.030**	**0.715 ± 0.037**	**0.715 ± 0.037**
MDI using gradient boosting	0.666 ± 0.016	0.666 ± 0.016	0.673 ± 0.018
Maximum depth limitation	0.625 ± 0.004	0.639 ± 0.023	0.687 ± 0.025

6 Conclusion

The objective of this paper was to devise an approach to minimize the number of survey questions needed to predict a patient's PTSD diagnosis. We demonstrated that it was possible to predict PTSD with about 72% accuracy using ≤10 features or 68% accuracy using ≤5 features, both of which are comparable to and possibly exceed the prediction accuracy of a classifier trained using the full survey data. We found that an approach based on optimizing the mean decrease in impurity (MDI) from a random forest model resulted in the highest accuracy.

While this work provides a promising start to PTSD prediction using minimized surveys, the overall prediction accuracy remains somewhat weak. This may be partially due to using only survey data taken within 2 weeks post-trauma to predict the PTSD diagnosis at 3 months post-trauma. In future work, we intend to incorporate the longitudinal data collected in the study to improve the overall prediction accuracy using multiple adaptive surveys that may ask different questions to different participants based on their past responses.

Acknowledgements. Research reported in this publication was supported by the National Institute of Mental Health of the National Institutes of Health under Award Numbers R21MH125277 and R01MH110483. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health.

References

1. Aha, D.W., Bankert, R.L.: A comparative evaluation of sequential feature selection algorithms. In: Pre-proceedings of the Fifth International Workshop on Artificial Intelligence and Statistics, pp. 1–7 (1995)
2. American Psychiatric Association: Diagnostic and statistical manual of mental disorders. 5th edn. (2013)
3. Bernstein, D.P., et al.: Development and validation of a brief screening version of the childhood trauma questionnaire. Child Abus. Negl. **27**(2), 169–190 (2003)
4. Breiman, L.: Random forests. Mach. Learn. **45**, 5–32 (2001)
5. Bryant, R.A., Moulds, M.L., Guthrie, R.M.: Acute stress disorder scale: a self-report measure of acute stress disorder. Psychol. Assess. **12**(1), 61 (2000)

6. van Buuren, S., Groothuis-Oudshoorn, K.: mice: Multivariate imputation by chained equations in R. J. Stat. Softw. **45**, 1–67 (2011)
7. Chen, J., et al.: Dispositional optimism mediates relations between childhood maltreatment and PTSD symptom severity among trauma-exposed adults. Child Abus. Negl. **115**, 105023 (2021)
8. Forbes, C.N., et al.: Emotional avoidance and social support interact to predict depression symptom severity one year after traumatic exposure. Psychiatry Res. **284**, 112746 (2020)
9. Friedman, J.H.: Greedy function approximation: a gradient boosting machine. Ann. Stat. **29**(5), 1189–1232 (2001)
10. Germain, A., Hall, M., Krakow, B., Shear, M.K., Buysse, D.J.: A brief sleep scale for posttraumatic stress disorder: Pittsburgh sleep quality index addendum for PTSD. J. Anxiety Disord. **19**(2), 233–244 (2005)
11. Gratz, K.L., Roemer, L.: Multidimensional assessment of emotion regulation and dysregulation: Development, factor structure, and initial validation of the difficulties in emotion regulation scale. J. Psychopathol. Behav. Assess. **26**, 41–54 (2004)
12. Huffman, N., et al.: Association of age of adverse childhood experiences with thalamic volumes and post-traumatic stress disorder in adulthood. Front. Behav. Neurosci. **17**, 1147686 (2023)
13. Jayachandran, S., Biradavolu, M., Cooper, J.: Using machine learning and qualitative interviews to design a five-question survey module for women's agency. World Dev. **161**, 106076 (2023)
14. Kshirsagar, V., Wieczorek, J., Ramanathan, S., Wells, R.: Household poverty classification in data-scarce environments: a machine learning approach (2017). arXiv preprint arXiv:1711.06813
15. Logé, F., Le Pennec, E., Amadou-Boubacar, H.: Intelligent questionnaires using approximate dynamic programming. i-com **19**(3), 227–237 (2021)
16. Louppe, G., Wehenkel, L., Sutera, A., Geurts, P.: Understanding variable importances in forests of randomized trees. In: Advances in Neural Information Processing Systems, vol. 26 (2013)
17. McCracken, L.M., Dhingra, L.: A short version of the pain anxiety symptoms scale (PASS-20): preliminary development and validity. Pain Res. Manag. **7**, 517163 (2002)
18. Meinshausen, N., Bühlmann, P.: Stability selection. J. Royal Stat. Soc. Ser. B Stat. Methodol. **72**(4), 417–473 (2010)
19. Nembrini, S., König, I.R., Wright, M.N.: The revival of the Gini importance? Bioinformatics **34**(21), 3711–3718 (2018)
20. Pedregosa, F., et al.: Scikit-learn: Machine learning in Python. J. Mach. Learn. Res. **12**, 2825–2830 (2011)
21. Rush, A., et al.: The 16-item quick inventory of depressive symptomatology (QIDS), clinician rating (QIDS-C), and self-report (QIDS-SR): a psychometric evaluation in patients with chronic major depression. Biol. Psychiatry **54**(5), 573–583 (2003)
22. Shih, C.H., et al.: Preliminary study examining the mediational link between mild traumatic brain injury, acute stress, and post-traumatic stress symptoms following trauma. Eur. J. Psychotraumatology **11**(1), 1815279 (2020)
23. Shih, C.H., Zhou, A., Grider, S., Xie, H., Wang, X., Elhai, J.D.: Early self-reported post-traumatic stress symptoms after trauma exposure and associations with diagnosis of post-traumatic stress disorder at 3 months: latent profile analysis. BJPsych Open **9**(1), e27 (2023)

24. Tibshirani, R.: Regression shrinkage and selection via the lasso. J. R. Stat. Soc. Ser. B Stat. Methodol. **58**(1), 267–288 (1996)
25. Xie, H., Shih, C.H., Aldoohan, S.D., Wall, J.T., Wang, X.: Hypothalamus volume mediates the association between adverse childhood experience and PTSD development after adulthood trauma. Transl. Psychiatry **13**(1), 274 (2023)
26. Zhou, A., McDaniel, M., Hong, X., Mattin, M., Wang, X., Shih, C.H.: Emotion dysregulation mediates the association between acute sleep disturbance and later posttraumatic stress symptoms in trauma exposed adults. Eur. J. Psychotraumatology **14**(2), 2202056 (2023)
27. Zou, H., Hastie, T.: Regularization and variable selection via the elastic net. J. R. Stat. Soc. Ser. B Stat. Methodol. **67**(2), 301–320 (2005)

Patient-Centric Approach for Utilising Machine Learning to Predict Health-Related Quality of Life Changes During Chemotherapy

Zuzanna Wójcik[1,2](✉), Vania Dimitrova[2], Lorraine Warrington[3], Galina Velikova[3,4], and Kate Absolom[3,5]

[1] UKRI Centre for Doctoral Training in Artificial Intelligence for Medical Diagnosis and Care, Leeds, UK
sczw@leeds.ac.uk
[2] School of Computing, University of Leeds, Leeds, UK
[3] Leeds Institute of Medical Research, University of Leeds, St James's University Hospital, Leeds, UK
[4] Leeds Cancer Centre, Leeds Teaching Hospitals NHS Trust, Leeds, UK
[5] Leeds Institute of Health Sciences, University of Leeds, Leeds, UK

Abstract. Patients undergoing chemotherapy often experience adverse effects, which can lead to changes in health-related quality of life (HRQOL) and have detrimental effects on patients' physical and psychological wellbeing. This study aims to apply machine learning (ML) models to patient-reported, clinical, and demographic data to predict changes in physical well-being, social functioning, role functioning, usual activities, and mobility at 6, 12 and 18 weeks from starting chemotherapy. A patient-centric approach is followed as outcome variables were selected after consultation with patients and a clinician, who also was involved in the study design. Logistic regression, random forest, extreme gradient boosting, and multilayer perceptron were developed and their performance of predicting improvement and deterioration in HRQOL was evaluated with accuracy, recall, specificity, and area under the ROC curve (AUC). Model performance was generally better when predicting improvement, with best models giving AUC of 0.904 for predicting mobility improvement at 12 weeks and AUC of 0.898 for predicting usual activities improvement at 18 weeks. The results encourage involving stakeholders in research and support the view that ML can be used to predict outcomes meaningful to patients. They also highlight that although some outcome variables can be valuable for patients, they may not be predicted well by ML models. This study can inform future work on patient-centric ML methods contributing to treatment decisions in oncology.

Keywords: Machine Learning · Patient-Centric Approach · Patient-Reported Data · Cancer Outcomes · Health-Related Quality Of Life

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 101–116, 2024.
https://doi.org/10.1007/978-3-031-66538-7_12

1 Introduction

Cancer diagnosis can be devastating for an individual and lead to emotional distress, anxiety or depression. Patients are usually faced with the immediate need to make a treatment decision, which is a daunting and complicated process due to serious adverse effects of chemotherapy. Often the treatment choice requires compromising between the quality of life and the length of life [1]. There are many existing studies on cancer survival analysis, but health-related quality of life (HRQOL) changes require more consideration in literature. Understanding factors related with increased chemotherapy toxicity and HRQOL fluctuations could personalise cancer care through enabling informed shared decision-making process and early preparation for potential adverse effects [2].

Machine learning (ML) has been successful in predicting patient outcomes in healthcare [3], but ML models are often built on clinical and demographic data, overlooking patients' views. Patient-reported outcome measures (PROMs) are questionnaires collecting patients' perception on their own health status, unaffected by clinical opinion [4]. PROMs added as input data in ML models have a potential to improve their predictive performance [5]. Furthermore, patients' perspective is often missed from the study design process. Therefore, the development of new ML models might not serve its purpose or address needs of diverse groups of people in an equal and fair way. Patient-centric research is also crucial for building public trust in artificial intelligence and support the implementation of the studied tools in clinical practice [6].

There is existing evidence that ML can predict chemotherapy toxicity, but research papers often focus on acute hospital utilisation during treatment, rather than HRQOL [7]. Studies foreseeing HRQOL during chemotherapy tend to use statistical modelling, rather than explore ML predictive opportunities [8]. Furthermore, the outcomes are usually limited to specific time during the chemotherapy, without investigating the changes from early to late stages of the treatment [9]. Patient and clinician involvement is also missing from the study design.

This study is a part of a broader research on using PROMs and ML to predict patient outcomes. In earlier work we have shown that baseline PROMs can predict hospital utilisation and treatment management [7]. In this paper we *focus on predicting patients' quality of life changes during chemotherapy following a patient-centric study design*. The study utilises PROMs in ML models to predict changes in physical wellbeing, social functioning, role functioning, usual activities, and mobility at 6, 12 and 18 weeks of chemotherapy. The paper ensures rigorous reporting of model development and evaluation, supporting reproducibility of research [10]. Stakeholder engagement was an essential part of the study design and model evaluation.

2 Methods

2.1 Overall Methodology

Four machine learning models: logistic regression (LR), random forest (RF), extreme gradient boosting (XGB), and multilayer perceptron (MLP) were used to predict patient-reported HRQOL changes at 6, 12 and 18 weeks of starting chemotherapy treatment. The overall study design is presented in Fig. 1.

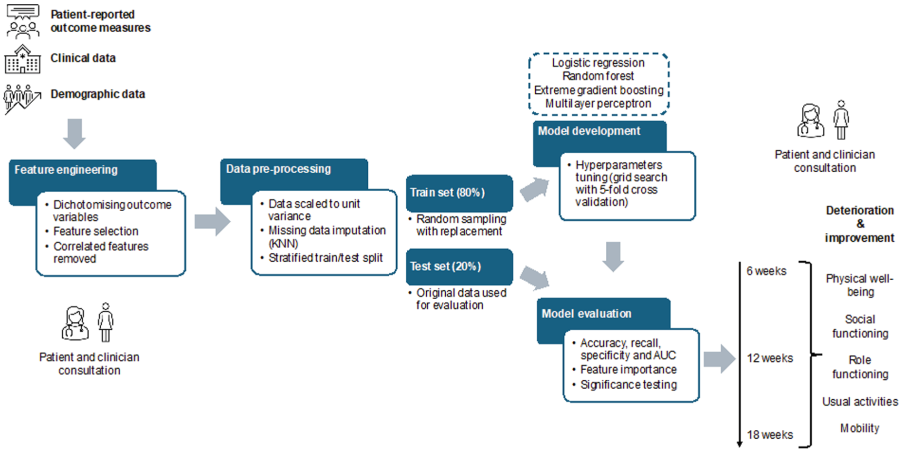

Patient-reported outcome measures
Clinical data
Demographic data
Feature engineering
• Dichotomising outcome variables
• Feature selection
• Correlated features removed
Patient and clinician consultation
Data pre-processing
• Data scaled to unit variance
• Missing data imputation (KNN)
• Stratified train/test split
Train set (80%)
• Random sampling with replacement
Test set (20%)
• Original data used for evaluation
Logistic regression
Random forest
Extreme gradient boosting
Multilayer perceptron
Model development
• Hyperparameters tuning (grid search with 5-fold cross validation)
Model evaluation
• Accuracy, recall, specificity and AUC
• Feature importance
• Significance testing
Patient and clinician consultation
Deterioration & improvement
6 weeks
12 weeks
18 weeks
Physical well-being
Social functioning
Role functioning
Usual activities
Mobility

Fig. 1. Flow diagram illustrating the overall methodology of the study.

2.2 Patient-Centric Approach

A consultation with patient representatives was a part of a Data-driven Cancer Research Conference 2024 in Manchester, UK, where initial study design was presented. Two of the patient representatives were asked which chemotherapy effects would be the most helpful to be aware of before treatment decisions. Patients stressed that apart from physical symptoms, they would like to understand how chemotherapy would impact their every-day activities and social life. Consequently, these aspects were selected as outcome variables in this study.

The patients' perspective was further considered, as a clinical oncologist was involved in all stages of the study. Patient Centred Outcomes Research (PCOR) Group at the Leeds Institute of Medical Research, University of Leeds, St James's University Hospital, a multidisciplinary research group, which includes oncologists, nurses and psychologists, was also consulted during earlier project stages. Following discussions with the group, the study includes feature importance analysis and ML model evaluation on original (unprocessed) data.

2.3 Dataset

The dataset contains data from 508 patients initiating chemotherapy for colorectal, breast, or gynecological cancers at Leeds Cancer Centre, collected in a prospective, randomized two-arm parallel group study, called eRAPID clinical trial [11]. It consists of 90 variables, 42 of which are demographic (self-reported) and clinical (from Electronic Healthcare Records). They include age at study entry, sex, marital status, level of education, employment status, body mass index (BMI), disease site (breast/gynecological/colorectal), previous chemotherapy (yes/no), disease type (metastatic/non-metastatic), and comorbidities presence: cardiovascular, respiratory, gastrointestinal, stomach/intestine, endocrine, renal, neurological, rheumatologic, previous malignancy, substance abuse. The remaining 48 variables consist of PROMs, completed by participants at 4 timepoints (baseline at the start of the trial, and at 6-, 12-, and 18-week follow-up). For each time-point, 6 variables were from Five-dimensional Visual Analogue Scale (EQ-5D-VAS [12]), including self-reported data on mobility, self-care, usual activities, pain/discomfort, anxiety/depression, and self-rated health status; 4 variables from Functional Assessment of Cancer Therapy - General 28 items (FACT-G [13]), including aggregated scores of physical, social, emotional and functional well-being; and 2 remaining variables were social and role scale from EORTC Core Quality of Life Questionnaire (QLQ-C30 [14]).

2.4 Feature Engineering

The outcome variables were improvement and deterioration in: physical wellbeing (physical symptoms), social functioning (ability to engage in the society), role functioning (ability to perform life roles), usual activities (ability to perform daily life activities), and mobility (ability to move) at 6, 12 and 18 weeks. Baseline values were subtracted from the values at the predicted time point. The improvement and deterioration were transformed into binary variables with the threshold of minimally clinically important difference (MCID) [15, 16] for physical well-being, social functioning and role functioning to ensure clinical relevance. Usual activities and mobility were 5-level, 1-item scale (from no problems to extreme problems) and have no MCID recommendations, so the deterioration and improvement of these variables were considered any change of ≤ 1 and ≤ -1 respectively (Table 1). Input features are described in *Dataset* subsection. Features correlated with Pearson coefficient higher than 0.6 were removed (leaving one) to ensure ML models process information efficiently. The list of removed variables in each feature set is presented in the Appendix Table 2.

2.5 Data Pre-processing

Continuous variables were standardised. Only rows (participants) with complete target variable at the predicted time point were included in the analysis. Any missing data were imputed using KNN imputer ($k = 5$) [17]. The number of missing data for each variable in each outcome is presented in Appendix Table 3. The data was split to train (80%) and test (20%) sets with stratification. Random sampling with replacement was performed on train set to ensure that models are not biased towards one class [18]. Test set was left imbalanced and models were evaluated on unprocessed data to have a potential to be applied to clinical practice. Original class distribution is provided in Table 1.

2.6 Model Development and Evaluation

LR, RF, XGB and MLP were applied using Python sklearn library to predict each target variable at each time point. The model selection was based on their frequency of use in studies applying ML to PROMs data to enable between studies comparison. Hyperparameters were tuned on train sets through grid search with five-fold cross-validation. The models were evaluated with accuracy, recall (also known as sensitivity), specificity and area under the ROC curve (AUC). Feature importance was also conducted on LR and RF models predicting 3 best performing outcomes at 18 weeks due to their explainability potential. Furthermore, Analysis of Variance (ANOVA) with Tukey's Honest Significant Difference (Tukey HSD) tests were performed to compare model performances.

Table 1. Predicted variables, questionnaires completed, change calculations, participants number (N), including deterioration (D) and improvement (N) cases.

HRQOL	Questionnaire	Deterioration	Improvement	Time point (N, D, I)
Physical well-being	Fact-G	change ≤ -2	change ≥ 2	6 weeks (N=439, D=252, I=68) 12 weeks(N=400, D=241, I=51) 18 weeks (N=382, D=219, I=58)
Social functioning	QLQ-C30	change ≤ -7	change ≥ 8	6 weeks (N=440, D=187, I=101) 12 weeks (N=407, D=209, I=73) 18 weeks (N=388, D=179, I=90)
Role functioning	QLQ-C30	change ≤ -6	change ≥ 11*	6 weeks (N=438, D=190, I=100) 12 weeks (N=407, D=216, I=90) 18 weeks (N=385, D=171, I=78)
Usual activities	EQ-5D-VAS	change ≥ 1	change ≥ -1	6 weeks (N=439, D=148, I=71) 12 weeks (N=408, D=181, I=50) 18 weeks (N=384, D=183, I=51)
Mobility	EQ-5D-VAS	change ≥ 1	change ≥ -1	6 weeks (N=438, D=64, I=48) 12 weeks (N=408, D=98, I=40) 18 weeks (N=382, D=122, I=31)

*MCID for role functioning improvement was the average from all cancer types due to availability [16].

3 Results

Accuracy, recall, specificity and AUC of all models are provided in Appendix Tables 4 and 5. The plots in Fig. 2 represent AUCs of all target variables apart from role functioning, due to its similarity to usual activities, but worse performance.

3.1 Predicted HRQOL Changes

Physical Well-Being. *Deterioration:* LR models had the highest AUCs at 6 (0.710), 12 (0.661) and 18 (0.761) weeks deterioration predictions. Even though LR did not provide the highest recall, it did not compromise the specificity unlike other models (e.g., MLP at 6 weeks: recall = 0.902, specificity = 0.162), preventing false negative predictions. All models had the best performance predicting deteriorations at 18 weeks, compared to 6 and 12 weeks. *Improvement:* MLP had the highest AUC overall at 12 weeks (0.843). This model also had high recall (0.800) and specificity (0.886). LR had the highest performance at 6 (AUC = 0.718) and 18 weeks (AUC = 0.662). Generally all models had poor recall (risking false improvement predictions), apart from LR (0.800) and aforementioned MLP (0.800) predicting improvement at 12 weeks. The models performed the best at 12 weeks, except for RF achieving highest AUC at 6 weeks.

Social Functioning. *Deterioration:* Overall, the best performing model was LR with highest AUC of 0.685 at 18 weeks. MLP was the best classifier at 6 weeks (AUC = 0.595), and LR at 12 weeks (AUC = 0.562). The models were compromising either specificity or recall, with at least one of them being lower than 0.600. Overall, 18 weeks deterioration predictions had highest AUC for all models except MLP (highest AUC of 0.595 at 6 weeks). *Improvement:* The best performing model was MLP at 6 (AUC = 0.713), 12 (AUC = 0.533) and 18 (AUC = 0.736) weeks. These models also had highest recall, which was 0.750 at 6 weeks and 0.722 at 18 weeks with specificity being respectively 0.676 and 0.750. The MLP's recall at 12 weeks was only 0.200, suggesting a bad performance of this model. RF had the highest specificity for all time points, but compromised recall. All models achieved best predictive performance at 18 weeks.

Role Functioning. *Deterioration:* LR outperformed other models, achieving the highest AUCs at all time points: 6 (0.573), 12 (0.691) and 18 (0.675) weeks. Recalls of these models were higher than specificity, with the highest value at 12 weeks (0.750). The models had best overall performance at 18 weeks, even though LR was slightly better at 12 weeks. *Improvement:* LR had the highest AUCs of all models at 6 (0.531) and 18 weeks (0.770), whilst MLP at 12 weeks (0.712). LR's performance at 18 weeks was the highest overall. This model also had the highest recall (0.688), but specificity of 0.852 was lower than XGB's specificity (0.902). Overall, models performed the best at 18 weeks (except MLP with the highest AUC = 0.712 at 12 weeks) and the worst at 6 weeks.

Usual Activities. *Deterioration:* The best performing model was LR at 6 (AUC = 0.663) and 12 weeks (AUC = 0.739), and MLP (AUC = 0.718) at 18 weeks. These models also had highest recall, but not specificity, which was the highest for RF (6 and 12 weeks) and XGB (18 weeks). In general, specificity was a lot higher than recall. The models had the lowest performance at 6 weeks, while 12 and 18 weeks predictions had similar performance. *Improvement:* LR outperformed all models at all time points with AUCs of 0.684 (6 weeks), 0.731 (12 weeks), and 0.898 (18 weeks). LR at 18 weeks was the highest performing model with excellent recall (0.900) and specificity (0.896). The specificity of all models was high, but often compromised recall. Models predicting 18 weeks improvement performed much better than models at other time points.

Mobility. *Deterioration:* The models had poor overall performance with MLP having the highest AUCs at 6 (0.529) and 12 (0.562) weeks, and LR (0.597) at 18 weeks. Recall of the models was poor as well, with the highest value of 0.560. Specificity values of the models were good overall, with the highest for RF at 6 weeks (0.973). It is unclear which time point resulted in the best predictions. *Improvement:* MLP was the best performing model overall achieving AUC of 0.904 (recall = 0.875, specificity = 0.932) at 12 weeks. LR was the best models at 6 weeks (AUC = 0.829) with very good recall (0.800) and specificity (0.859). At 18 weeks LR also outperformed other models with AUC of 0.715, high specificity (0.930), but low recall (0.500). Overall, 18 weeks predictions resulted in the lowest AUCs, while 6- and 12-week predictions depended on the model.

3.2 Significance Testing

ANOVA results suggest that at least one model's AUC mean is different from the rest (F = 4.202, p = 0.007) and Tukey HSD indicated that LR was significantly better than RF (0.011) and XGB (0.041). Outcome also affected prediction performance (F = 4.659, p = 0.002), as predictions of usual activities had higher mean AUC than role (p = 0.0364) and social (p = 0.001) functioning. The time point of prediction was also a factor affecting AUC (F = 5.676, p = 0.004), with Tukey HSD indicating 18-weeks predictions resulting with significantly higher mean AUC than 6-weeks predictions (p = 0.003). Welch Two Sample t-test indicated that improvement predictions were significantly better than deterioration predictions (t = -3.079, df = 101.39, p = 0.003).

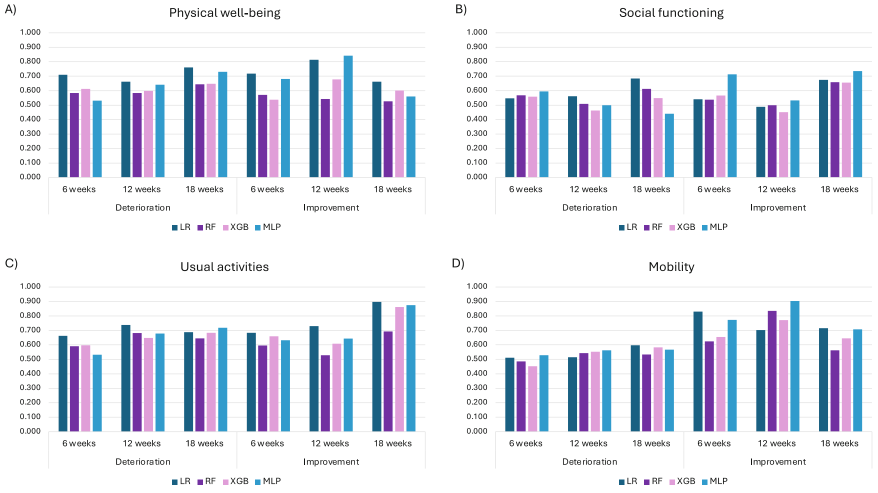


Fig. 2. AUC values of models predicting A) physical well-being, B) social functioning, C) usual activities, D) mobility deterioration and improvement at 6, 12, and 18 weeks.

3.3 Feature Importance

Features with coefficients for LR and values of importance for RF are presented in Appendix Tables 6, 7, 8. For prediction of physical well-being, LR looked mainly at clinical information (comorbidities and cancer characteristics). RF considered Fact-G and EQ-5D as most important predictors, as well as patients' BMI. LR in predicting mobility also found comorbidities the most meaningful predictors (including EQ5D mobility score at baseline for improvement prediction), while RF mainly looked at PROMs, cancer type and BMI. When predicting usual activities, LR also looked at the clinical characteristics and usual activities baseline score, whilst RF focused on BMI, cancer type and PROMs.

4 Discussion

4.1 Discussion of the Study Findings

Overall results. The findings suggest that ML models applied on PROMs, clinical and demographic data can successfully predict HRQOL outcomes throughout chemotherapy which are meaningful to patients. Models provided excellent performance in predicting improvement in physical well-being, usual activities and mobility at different stages of cancer treatment. Consultation with a clinical oncologist endorsed the view that ML prediction of HRQOL changes during chemotherapy can be useful in clinical practice.

Model performances. LR generally outperformed other models, which is a common outcome in medical research, as other models are more susceptible to overfitting [19]. In some cases, MLP achieved higher performance than LR. Nevertheless, the lack of explainability of MLP could affect public trust in this model. According to the consulted clinical oncologist, even impossible to interpret models should be considered as useful, as long as they are used alongside other well-performing models, which enable explainability.

Change at given time points. Improvement predictions had generally higher performance than deterioration, even though chemotherapy is associated with decline in HRQOL [8]. However, the deterioration might depend on patient characteristics from the start of chemotherapy. For example, metastatic disease may be more likely to show improvement due to higher burden of cancer symptoms prior to treatment. These differences will be further explored.

4.2 Strengths and Limitations

The main strength of this work is the patient-centric approach achieved through the active engagement of a clinical oncologist, consultation with patient representatives and patient reports used as input data. Three time points of HRQOL changes provided another insight into fluctuation of chemotherapy symptoms and when they can affect individuals. Finally, the rigorous reporting of data pre-processing methods, model development and evaluation supports the reproducibility of this study. However, the ethical approval does not allow data sharing, which might negatively impact the reproducibility. Furthermore, this study has limitations typical for data collected in clinical trials. While the ML models have been rigorously designed and evaluated, the data were subject to inclusion and exclusion criteria, which can lead to bias. Temporal clinical trial data are usually relatively small samples and are affected by participants drop outs. This limits the ML methods that have been applied, yet the results are encouraging.

4.3 Conclusions and Future Work

This study successfully applied ML models on PROMs, clinical and demographic data to predict changes in HRQOL during chemotherapy, which could support preparation for adverse effects of chemotherapy and inform treatment decisions. The results further encourage the use of ML methods to identify factors related to chemotherapy toxicity and explore how cancer treatment affects individuals' lives. Patient and clinician involvement ensured that the predicted variables are meaningful for patients and clinically relevant. We are currently extending the stakeholder engagement by designing ways to explain ML models and evaluate possible clinical adoption of the findings from a more representative group of patients. We are also using longitudinally collected PROMs and symptom reports for patient outcome predictions. This will consider traditional ML models and deep learning methods to process multi-dimensional time-series data.

Acknowledgements. The authors thank the patients and clinicians participating in the eRAPID clinical trial for providing the data; Patient Centred Outcomes Research (PCOR) Group in the University of Leeds, Faculty of Medicine and Health, Leeds Institute of Medical Research for consultations of initial study design; and patient representatives of the Use My Data for discussing research relevance. This work was supported in part by UK Research and Innovation (UKRI) [CDT grant number EP/S024336/1].

Appendix

Table 2. Removed features based on correlation analysis.

Physical well-being	6 weeks	'SOC0', 'ROL0', 'AgeStudyEntry', 'Sex', 'EWB_overall0'
	12 weeks	FWB_overall0', 'SOC0', 'ROL0', 'AgeStudyEntry', 'Sex', 'EWB_overall0', 'CMRheuCTD'
	18 weeks	AgeStudyEntry', 'EWB_overall0', 'Sex', 'SOC0'
Social functioning	6 weeks	SOC0', 'EWB_overall0', 'AgeStudyEntry', 'ROL0', 'Sex'
	12 weeks	SOC0', 'AgeStudyEntry', 'EWB_overall0', 'CMRheuCTD', 'CMCarHyperten', 'FWB_overall0', 'ROL0', 'Sex'
	18 weeks	SOC0', 'EWB_overall0', 'AgeStudyEntry', 'Sex'
Role functioning	6 weeks	SOC0', 'ROL0', 'Sex', 'AgeStudyEntry', 'FWB_overall0', 'EWB_overall0'
	12 weeks	SOC0', 'ROL0', 'Sex', 'CMRheuCTD', 'AgeStudyEntry', 'FWB_overall0', 'EWB_overall0', 'CMCarHyperten'
	18 weeks	SOC0', 'Sex', 'AgeStudyEntry', 'EWB_overall0', 'CMCarHyperten'
Usual activities	6 weeks	ROL0', 'EWB_overall0', 'FWB_overall0', 'Sex', 'AgeStudyEntry', 'SOC0'
	12 weeks	'ROL0', 'EWB_overall0', 'CMCarHyperten', 'FWB_overall0', 'CMRheuCTD', 'Sex', 'AgeStudyEntry', 'SOC0'
	18 weeks	EWB_overall0', 'FWB_overall0', 'Sex', 'AgeStudyEntry', 'SOC0'
Mobility	6 weeks	'AgeStudyEntry', 'PWB_overall0', 'SOC0', 'ROL0', 'Sex', 'FWB_overall0', 'EWB_overall0'
	12 weeks	'AgeStudyEntry', 'ROL0', 'SOC0', 'Sex', 'CMCarHyperten', 'FWB_overall0', 'CMRheuCTD', 'EWB_overall0'
	18 weeks	'AgeStudyEntry', 'SOC0', 'ROL0', 'Sex', 'EWB_overall0'

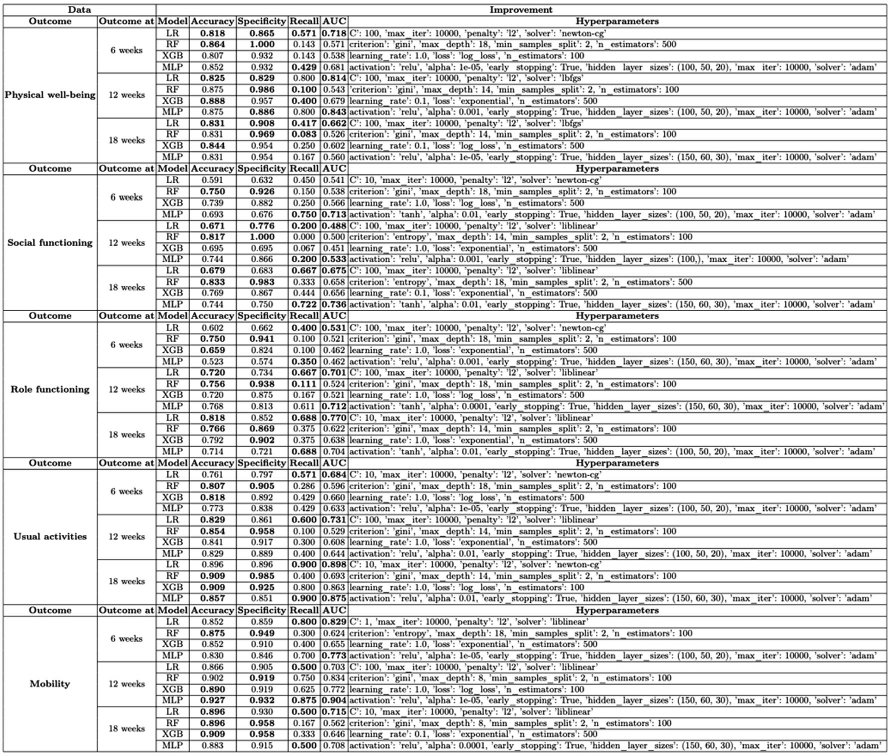
Table 3. Number of missing data in each variable for each outcome at 18 weeks.

Physical well-being		Social functioning		Role functioning		Usual activities		Mobility	
StudyArm	0	StudyArm	0	StudyArm	0	StudyArm	0	StudyArm	0
DiseaseSite	0	DiseaseSite	0	DiseaseSite	0	DiseaseSite	0	DiseaseSite	0
Sex	0	Sex	0	Sex	0	Sex	0	Sex	0
PreviousChemo	0	PreviousChemo	0	PreviousChemo	0	PreviousChemo	0	PreviousChemo	0
AgeStudyEntry	0	AgeStudyEntry	0	AgeStudyEntry	0	AgeStudyEntry	0	AgeStudyEntry	0
PrimaryorMet	0	PrimaryorMet	0	PrimaryorMet	0	PrimaryorMet	0	PrimaryorMet	0
BCBMI	1	BCBMI	1	BCBMI	1	BCBMI	1	BCBMI	2
Comorbidities	0	Comorbidities	0	Comorbidities	0	Comorbidities	0	Comorbidities	0
CMCarMI	0	CMCarMI	0	CMCarMI	0	CMCarMI	0	CMCarMI	0
CMCarAngina	0	CMCarAngina	0	CMCarAngina	0	CMCarAngina	0	CMCarAngina	0
CMCarHeartFail	0	CMCarHeartFail	0	CMCarHeartFail	0	CMCarHeartFail	0	CMCarHeartFail	0
CMCarArrhythm	0	CMCarArrhythm	0	CMCarArrhythm	0	CMCarArrhythm	0	CMCarArrhythm	0
CMCarHyperten	0	CMCarHyperten	0	CMCarHyperten	0	CMCarHyperten	0	CMCarHyperten	0
CMCarVenous	0	CMCarVenous	0	CMCarVenous	0	CMCarVenous	0	CMCarVenous	0
CMResCOPD	0	CMResCOPD	0	CMResCOPD	0	CMResCOPD	0	CMResCOPD	0
CMResEmphys	0	CMResEmphys	0	CMResEmphys	0	CMResEmphys	0	CMResEmphys	0
CMResAsthma	0	CMResAsthma	0	CMResAsthma	0	CMResAsthma	0	CMResAsthma	0
CMResChronBron	0	CMResChronBron	0	CMResChronBron	0	CMResChronBron	0	CMResChronBron	0
CMGasChronHep	0	CMGasChronHep	0	CMGasChronHep	0	CMGasChronHep	0	CMGasChronHep	0
CMGasCirrhosis	0	CMGasCirrhosis	0	CMGasCirrhosis	0	CMGasCirrhosis	0	CMGasCirrhosis	0
CMGasPancreas	0	CMGasPancreas	0	CMGasPancreas	0	CMGasPancreas	0	CMGasPancreas	0
CMStomUlcers	0	CMStomUlcers	0	CMStomUlcers	0	CMStomUlcers	0	CMStomUlcers	0
CMStomMalabsor	0	CMStomMalabsor	0	CMStomMalabsor	0	CMStomMalabsor	0	CMStomMalabsor	0
CMStomInflamm	0	CMStomInflamm	0	CMStomInflamm	0	CMStomInflamm	0	CMStomInflamm	0
CMEndDiabetes	0	CMEndDiabetes	0	CMEndDiabetes	0	CMEndDiabetes	0	CMEndDiabetes	0
CMEndHypothy	0	CMEndHypothy	0	CMEndHypothy	0	CMEndHypothy	0	CMEndHypothy	0
CMEndHyperth	0	CMEndHyperth	0	CMEndHyperth	0	CMEndHyperth	0	CMEndHyperth	0
CMRenEndStage	0	CMRenEndStage	0	CMRenEndStage	0	CMRenEndStage	0	CMRenEndStage	0
CMNeuStroke	0	CMNeuStroke	0	CMNeuStroke	0	CMNeuStroke	0	CMNeuStroke	0
CMNeuMS	0	CMNeuMS	0	CMNeuMS	0	CMNeuMS	0	CMNeuMS	0
CMNeuParkins	0	CMNeuParkins	0	CMNeuParkins	0	CMNeuParkins	0	CMNeuParkins	0
CMNeuMyasth	0	CMNeuMyasth	0	CMNeuMyasth	0	CMNeuMyasth	0	CMNeuMyasth	0
CMRheuArth	0	CMRheuArth	0	CMRheuArth	0	CMRheuArth	0	CMRheuArth	0
CMRheuLupus	0	CMRheuLupus	0	CMRheuLupus	0	CMRheuLupus	0	CMRheuLupus	0
CMRheuCTD	0	CMRheuCTD	0	CMRheuCTD	0	CMRheuCTD	0	CMRheuCTD	0
CMRheuPolymyo	0	CMRheuPolymyo	0	CMRheuPolymyo	0	CMRheuPolymyo	0	CMRheuPolymyo	0
CMRheuRhPolymy	0	CMRheuRhPolymy	0	CMRheuRhPolymy	0	CMRheuRhPolymy	0	CMRheuRhPolymy	0
CMPrevMal	0	CMPrevMal	0	CMPrevMal	0	CMPrevMal	0	CMPrevMal	0
CMSubstAlcohol	0	CMSubstAlcohol	0	CMSubstAlcohol	0	CMSubstAlcohol	0	CMSubstAlcohol	0
CMSubstDrugs	0	CMSubstDrugs	0	CMSubstDrugs	0	CMSubstDrugs	0	CMSubstDrugs	0
DCMarital	4	DCMarital	4	DCMarital	4	DCMarital	4	DCMarital	4
DCEmployment	12	DCEmployment	12	DCEmployment	12	DCEmployment	12	DCEmployment	12
ed_lev	12	ed_lev	12	ed_lev	12	ed_lev	12	ed_lev	12
EQ5DMob0	4	EQ5DMob0	4	EQ5DMob0	4	EQ5DMob0	3	EQ5DMob0	2
EQ5DSelCar0	3	EQ5DSelCar0	3	EQ5DSelCar0	3	EQ5DSelCar0	2	EQ5DMob18	0
EQ5DUsuAct0	3	EQ5DUsuAct0	3	EQ5DUsuAct0	3	EQ5DUsuAct0	2	EQ5DSelCar0	2
EQ5DPain0	5	EQ5DPain0	5	EQ5DPain0	5	EQ5DUsuAct18	0	EQ5DUsuAct0	2
EQ5DAnxDep0	5	EQ5DAnxDep0	5	EQ5DAnxDep0	5	EQ5DPain0	4	EQ5DPain0	4
EQ5DVAS0	4	EQ5DVAS0	4	EQ5DVAS0	4	EQ5DAnxDep0	4	EQ5DAnxDep0	4
ROL0	2	ROL0	2	ROL0	2	EQ5DVAS0	3	EQ5DVAS0	3
SOC0	2	SOC0	2	ROL18	0	ROL0	2	ROL0	3
PWB_overall0	8	SOC18	0	SOC0	2	SOC0	2	SOC0	3
PWB_overall18	0	PWB_overall0	8	PWB_overall0	8	PWB_overall0	8	PWB_overall0	9
SWB_overall0	4	SWB_overall0	4	SWB_overall0	4	SWB_overall0	4	SWB_overall0	5
EWB_overall0	1	EWB_overall0	1	EWB_overall0	1	EWB_overall0	1	EWB_overall0	2
FWB_overall0	1	FWB_overall0	1	FWB_overall0	1	FWB_overall0	1	FWB_overall0	2

Table 4. Deterioration prediction results with hyperparameters used for model development.

Data		Deterioration					
Outcome	**Outcome at**	**Model**	**Accuracy**	**Specificity**	**Recall**	**AUC**	**Hyperparameters**
Physical well-being	6 weeks	LR	**0.716**	**0.676**	0.745	**0.710**	C’: 10, ’max_iter’: 10000, ’penalty’: ’l2’, ’solver’: ’liblinear’
		RF	0.625	0.324	0.843	0.584	criterion’: ’gini’, ’max_depth’: 18, ’min_samples_split’: 2, ’n_estimators’: 500
		XGB	0.636	0.459	0.765	0.612	learning_rate’: 1.0, ’loss’: ’exponential’, ’n_estimators’: 500
		MLP	0.591	0.162	**0.902**	0.532	activation’: ’tanh’, ’alpha’: 1e-05, ’early_stopping’: True, ’hidden_layer_sizes’: (100, 50, 20), ’max_iter’: 10000, ’solver’: ’adam’
	12 weeks	LR	**0.663**	**0.656**	0.667	**0.661**	C’: 10, ’max_iter’: 10000, ’penalty’: ’l2’, ’solver’: ’liblinear’
		RF	0.613	0.438	**0.729**	0.583	criterion’: ’entropy’, ’max_depth’: 18, ’min_samples_split’: 2, ’n_estimators’: 500
		XGB	0.625	0.469	**0.729**	0.599	learning_rate’: 1.0, ’loss’: ’log_loss’, ’n_estimators’: 100
		MLP	0.638	**0.656**	0.625	0.641	activation’: ’relu’, ’alpha’: 0.01, ’early_stopping’: True, ’hidden_layer_sizes’: (150, 60, 30), ’max_iter’: 10000, ’solver’: ’adam’
	18 weeks	LR	**0.766**	**0.727**	0.795	**0.761**	C’: 1, ’max_iter’: 10000, ’penalty’: ’l2’, ’solver’: ’liblinear’
		RF	0.675	0.424	**0.864**	0.644	criterion’: ’entropy’, ’max_depth’: 14, ’min_samples_split’: 2, ’n_estimators’: 100
		XGB	0.662	0.545	0.750	0.648	learning_rate’: 0.1, ’loss’: ’exponential’, ’n_estimators’: 500
		MLP	0.740	0.667	0.795	0.731	activation’: ’relu’, ’alpha’: 0.001, ’early_stopping’: True, ’hidden_layer_sizes’: (150, 60, 30), ’max_iter’: 10000, ’solver’: ’adam’
Outcome	**Outcome at**	**Model**	**Accuracy**	**Specificity**	**Recall**	**AUC**	**Hyperparameters**
Social functioning	6 weeks	LR	0.557	0.608	0.486	0.547	C’: 1, ’max_iter’: 10000, ’penalty’: ’l2’, ’solver’: ’liblinear’
		RF	**0.602**	**0.784**	0.351	0.568	criterion’: ’entropy’, ’max_depth’: 14, ’min_samples_split’: 2, ’n_estimators’: 500
		XGB	0.591	0.765	0.351	0.558	learning_rate’: 1.0, ’loss’: ’log_loss’, ’n_estimators’: 500
		MLP	0.591	0.569	**0.622**	**0.595**	activation’: ’tanh’, ’alpha’: 1e-05, ’early_stopping’: True, ’hidden_layer_sizes’: (100, 50, 20), ’max_iter’: 10000, ’solver’: ’adam’
	12 weeks	LR	**0.561**	**0.600**	0.524	**0.562**	C’: 10, ’max_iter’: 10000, ’penalty’: ’l2’, ’solver’: ’lbfgs’
		RF	0.512	0.375	0.643	0.509	criterion’: ’entropy’, ’max_depth’: 18, ’min_samples_split’: 2, ’n_estimators’: 100
		XGB	0.463	0.425	0.500	0.463	learning_rate’: 0.1, ’loss’: ’log_loss’, ’n_estimators’: 500
		MLP	0.512	0.000	**1.000**	0.500	activation’: ’relu’, ’alpha’: 0.001, ’early_stopping’: True, ’hidden_layer_sizes’: (100, 50, 20), ’max_iter’: 10000, ’solver’: ’adam’
	18 weeks	LR	**0.692**	0.786	**0.583**	**0.685**	C’: 0.1, ’max_iter’: 10000, ’penalty’: ’l2’, ’solver’: ’liblinear’
		RF	0.628	**0.810**	0.417	0.613	criterion’: ’entropy’, ’max_depth’: 18, ’min_samples_split’: 2, ’n_estimators’: 500
		XGB	0.564	0.738	0.361	0.550	learning_rate’: 0.1, ’loss’: ’log_loss’, ’n_estimators’: 500
		MLP	0.436	0.381	0.500	0.440	activation’: ’tanh’, ’alpha’: 1e-05, ’early_stopping’: True, ’hidden_layer_sizes’: (100, 50, 20), ’max_iter’: 10000, ’solver’: ’sgd’
Outcome	**Outcome at**	**Model**	**Accuracy**	**Specificity**	**Recall**	**AUC**	**Hyperparameters**
Role functioning	6 weeks	LR	0.568	0.540	0.605	**0.573**	C’: 0.1, ’max_iter’: 10000, ’penalty’: ’l2’, ’solver’: ’newton-cg’
		RF	0.580	**0.820**	0.263	0.542	criterion’: ’entropy’, ’max_depth’: 18, ’min_samples_split’: 2, ’n_estimators’: 100
		XGB	**0.591**	0.780	0.342	0.561	learning_rate’: 1.0, ’loss’: ’log_loss’, ’n_estimators’: 500
		MLP	0.432	0.000	**1.000**	0.500	activation’: ’relu’, ’alpha’: 1e-05, ’early_stopping’: True, ’hidden_layer_sizes’: (100, 50, 20), ’max_iter’: 10000, ’solver’: ’sgd’
	12 weeks	LR	**0.695**	0.632	0.750	**0.691**	C’: 0.1, ’max_iter’: 10000, ’penalty’: ’l2’, ’solver’: ’liblinear’
		RF	0.646	**0.646**	**0.841**	0.631	criterion’: ’entropy’, ’max_depth’: 14, ’min_samples_split’: 2, ’n_estimators’: 100
		XGB	0.573	0.395	0.727	0.561	learning_rate’: 1.0, ’loss’: ’log_loss’, ’n_estimators’: 500
		MLP	0.524	0.395	0.636	0.516	activation’: ’relu’, ’alpha’: 0.01, ’early_stopping’: True, ’hidden_layer_sizes’: (100, 50, 20), ’max_iter’: 10000, ’solver’: ’adam’
	18 weeks	LR	**0.675**	0.674	0.676	**0.675**	C’: 0.1, ’max_iter’: 10000, ’penalty’: ’l2’, ’solver’: ’newton-cg’
		RF	**0.675**	**0.837**	0.471	0.654	criterion’: ’gini’, ’max_depth’: 14, ’min_samples_split’: 2, ’n_estimators’: 500
		XGB	0.636	0.721	0.529	0.625	learning_rate’: 1.0, ’loss’: ’exponential’, ’n_estimators’: 100
		MLP	0.662	0.628	**0.706**	0.667	activation’: ’tanh’, ’alpha’: 0.01, ’early_stopping’: True, ’hidden_layer_sizes’: (100,), ’max_iter’: 10000, ’solver’: ’adam’
Outcome	**Outcome at**	**Model**	**Accuracy**	**Specificity**	**Recall**	**AUC**	**Hyperparameters**
Usual activities	6 weeks	LR	0.693	0.759	**0.567**	**0.663**	C’: 1, ’max_iter’: 10000, ’penalty’: ’l2’, ’solver’: ’newton-cg’
		RF	**0.705**	**0.948**	0.233	0.591	criterion’: ’gini’, ’max_depth’: 18, ’min_samples_split’: 2, ’n_estimators’: 100
		XGB	0.670	0.828	0.367	0.597	learning_rate’: 1.0, ’loss’: ’log_loss’, ’n_estimators’: 500
		MLP	0.659	0.931	0.133	0.532	activation’: ’tanh’, ’alpha’: 1e-05, ’early_stopping’: True, ’hidden_layer_sizes’: (150, 60, 30), ’max_iter’: 10000, ’solver’: ’adam’
	12 weeks	LR	**0.744**	0.783	**0.694**	**0.739**	C’: 1, ’max_iter’: 10000, ’penalty’: ’l2’, ’solver’: ’liblinear’
		RF	0.707	**0.891**	0.472	0.682	criterion’: ’entropy’, ’max_depth’: 14, ’min_samples_split’: 2, ’n_estimators’: 100
		XGB	0.671	0.826	0.472	0.649	learning_rate’: 0.1, ’loss’: ’exponential’, ’n_estimators’: 500
		MLP	0.683	0.717	0.639	0.678	activation’: ’relu’, ’alpha’: 1e-05, ’early_stopping’: True, ’hidden_layer_sizes’: (100,), ’max_iter’: 10000, ’solver’: ’adam’
	18 weeks	LR	0.688	0.675	0.703	0.689	C’: 1, ’max_iter’: 10000, ’penalty’: ’l2’, ’solver’: ’newton-cg’
		RF	0.649	0.750	0.541	0.645	criterion’: ’gini’, ’max_depth’: 14, ’min_samples_split’: 2, ’n_estimators’: 500
		XGB	0.688	**0.800**	0.568	0.684	learning_rate’: 1.0, ’loss’: ’exponential’, ’n_estimators’: 500
		MLP	**0.714**	0.625	**0.811**	**0.718**	activation’: ’tanh’, ’alpha’: 0.01, ’early_stopping’: True, ’hidden_layer_sizes’: (100, 50, 20), ’max_iter’: 10000, ’solver’: ’adam’
Outcome	**Outcome at**	**Model**	**Accuracy**	**Specificity**	**Recall**	**AUC**	**Hyperparameters**
Mobility	6 weeks	LR	0.545	0.560	**0.462**	0.511	C’: 100, ’max_iter’: 10000, ’penalty’: ’l2’, ’solver’: ’liblinear’
		RF	**0.830**	**0.973**	0.000	0.487	criterion’: ’gini’, ’max_depth’: 18, ’min_samples_split’: 2, ’n_estimators’: 100
		XGB	0.716	0.827	0.077	0.452	learning_rate’: 1.0, ’loss’: ’log_loss’, ’n_estimators’: 500
		MLP	0.739	0.827	0.231	**0.529**	activation’: ’relu’, ’alpha’: 0.001, ’early_stopping’: True, ’hidden_layer_sizes’: (150, 60, 30), ’max_iter’: 10000, ’solver’: ’adam’
	12 weeks	LR	0.549	0.581	**0.450**	0.515	C’: 10, ’max_iter’: 10000, ’penalty’: ’l2’, ’solver’: ’newton-cg’
		RF	0.744	**0.935**	0.150	0.543	criterion’: ’gini’, ’max_depth’: 14, ’min_samples_split’: 2, ’n_estimators’: 500
		XGB	**0.683**	0.806	0.300	0.553	’learning_rate’: 1.0, ’loss’: ’exponential’, ’n_estimators’: 500
		MLP	0.671	0.774	0.350	**0.562**	activation’: ’relu’, ’alpha’: 0.0001, ’early_stopping’: True, ’hidden_layer_sizes’: (150, 60, 30), ’max_iter’: 10000, ’solver’: ’adam’
	18 weeks	LR	0.610	0.635	**0.560**	**0.597**	C’: 100, ’max_iter’: 10000, ’penalty’: ’l2’, ’solver’: ’lbfgs’
		RF	0.623	0.788	0.280	0.534	criterion’: ’gini’, ’max_depth’: 18, ’min_samples_split’: 2, ’n_estimators’: 100
		XGB	**0.662**	**0.808**	0.360	0.584	learning_rate’: 0.1, ’loss’: ’log_loss’, ’n_estimators’: 500
		MLP	0.584	0.615	0.520	0.568	activation’: ’relu’, ’alpha’: 1e-05, ’early_stopping’: True, ’hidden_layer_sizes’: (150, 60, 30), ’max_iter’: 10000, ’solver’: ’adam’

Table 5. Improvement prediction results with hyperparameters used for model development.



Data		Improvement					
Outcome	**Outcome at**	**Model**	**Accuracy**	**Specificity**	**Recall**	**AUC**	**Hyperparameters**
Physical well-being	6 weeks	LR	**0.818**	**0.865**	**0.571**	**0.718**	C’: 100, ’max_iter’: 10000, ’penalty’: ’l2’, ’solver’: ’newton-cg’
		RF	**0.864**	**1.000**	0.143	0.571	criterion’: ’gini’, ’max_depth’: 18, ’min_samples_split’: 2, ’n_estimators’: 500
		XGB	0.807	0.932	0.143	0.538	learning_rate’: 1.0, ’loss’: ’log_loss’, ’n_estimators’: 100
		MLP	0.852	0.932	**0.429**	0.681	activation’: ’relu’, ’alpha’: 1e-05, ’early_stopping’: True, ’hidden_layer_sizes’: (100, 50, 20), ’max_iter’: 10000, ’solver’: ’adam’
	12 weeks	LR	**0.825**	**0.829**	0.800	**0.814**	C’: 100, ’max_iter’: 10000, ’penalty’: ’l2’, ’solver’: ’lbfgs’
		RF	0.875	**0.986**	**0.100**	0.543	’criterion’: ’gini’, ’max_depth’: 14, ’min_samples_split’: 2, ’n_estimators’: 100
		XGB	**0.888**	0.957	**0.400**	0.679	learning_rate’: 0.1, ’loss’: ’exponential’, ’n_estimators’: 500
		MLP	0.875	**0.886**	0.800	**0.843**	activation’: ’relu’, ’alpha’: 0.001, ’early_stopping’: True, ’hidden_layer_sizes’: (100, 50, 20), ’max_iter’: 10000, ’solver’: ’adam’
	18 weeks	LR	**0.831**	**0.908**	**0.417**	**0.662**	C’: 100, ’max_iter’: 10000, ’penalty’: ’l2’, ’solver’: ’lbfgs’
		RF	0.831	**0.969**	**0.083**	0.526	criterion’: ’gini’, ’max_depth’: 14, ’min_samples_split’: 2, ’n_estimators’: 100
		XGB	**0.844**	0.954	0.250	0.602	learning_rate’: 0.1, ’loss’: ’log_loss’, ’n_estimators’: 500
		MLP	0.831	0.954	0.167	0.560	activation’: ’relu’, ’alpha’: 1e-05, ’early_stopping’: True, ’hidden_layer_sizes’: (150, 60, 30), ’max_iter’: 10000, ’solver’: ’adam’
Outcome	**Outcome at**	**Model**	**Accuracy**	**Specificity**	**Recall**	**AUC**	**Hyperparameters**
Social functioning	6 weeks	LR	0.591	0.632	0.450	0.541	C’: 10, ’max_iter’: 10000, ’penalty’: ’l2’, ’solver’: ’newton-cg’
		RF	**0.750**	**0.926**	0.150	0.538	criterion’: ’gini’, ’max_depth’: 18, ’min_samples_split’: 2, ’n_estimators’: 100
		XGB	0.739	0.882	0.250	0.566	learning_rate’: 1.0, ’loss’: ’log_loss’, ’n_estimators’: 500
		MLP	0.693	0.676	**0.750**	**0.713**	activation’: ’tanh’, ’alpha’: 0.01, ’early_stopping’: True, ’hidden_layer_sizes’: (100, 50, 20), ’max_iter’: 10000, ’solver’: ’adam’
	12 weeks	LR	**0.671**	**0.776**	**0.200**	**0.488**	C’: 100, ’max_iter’: 10000, ’penalty’: ’l2’, ’solver’: ’liblinear’
		RF	**0.817**	**1.000**	0.000	0.500	criterion’: ’entropy’, ’max_depth’: 14, ’min_samples_split’: 2, ’n_estimators’: 100
		XGB	0.695	0.695	0.067	0.451	learning_rate’: 1.0, ’loss’: ’exponential’, ’n_estimators’: 500
		MLP	0.744	0.866	**0.200**	**0.533**	activation’: ’relu’, ’alpha’: 0.001, ’early_stopping’: True, ’hidden_layer_sizes’: (100,), ’max_iter’: 10000, ’solver’: ’adam’
	18 weeks	LR	**0.679**	0.683	**0.667**	**0.675**	C’: 100, ’max_iter’: 10000, ’penalty’: ’l2’, ’solver’: ’liblinear’
		RF	**0.833**	**0.983**	0.333	0.658	criterion’: ’entropy’, ’max_depth’: 18, ’min_samples_split’: 2, ’n_estimators’: 500
		XGB	0.769	0.867	0.444	0.656	learning_rate’: 0.1, ’loss’: ’exponential’, ’n_estimators’: 500
		MLP	0.744	0.750	**0.722**	**0.736**	activation’: ’tanh’, ’alpha’: 0.01, ’early_stopping’: True, ’hidden_layer_sizes’: (150, 60, 30), ’max_iter’: 10000, ’solver’: ’adam’
Outcome	**Outcome at**	**Model**	**Accuracy**	**Specificity**	**Recall**	**AUC**	**Hyperparameters**
Role functioning	6 weeks	LR	0.602	0.662	**0.400**	**0.531**	C’: 100, ’max_iter’: 10000, ’penalty’: ’l2’, ’solver’: ’newton-cg’
		RF	**0.750**	**0.941**	0.100	0.521	criterion’: ’gini’, ’max_depth’: 18, ’min_samples_split’: 2, ’n_estimators’: 100
		XGB	**0.659**	0.824	0.100	0.462	learning_rate’: 1.0, ’loss’: ’exponential’, ’n_estimators’: 500
		MLP	0.523	0.574	**0.350**	0.462	activation’: ’relu’, ’alpha’: 0.001, ’early_stopping’: True, ’hidden_layer_sizes’: (150, 60, 30), ’max_iter’: 10000, ’solver’: ’adam’
	12 weeks	LR	**0.720**	0.734	**0.667**	**0.701**	C’: 100, ’max_iter’: 10000, ’penalty’: ’l2’, ’solver’: ’liblinear’
		RF	**0.756**	**0.938**	**0.111**	0.524	criterion’: ’gini’, ’max_depth’: 18, ’min_samples_split’: 2, ’n_estimators’: 100
		XGB	0.720	0.875	0.167	0.521	learning_rate’: 1.0, ’loss’: ’log_loss’, ’n_estimators’: 500
		MLP	0.768	0.813	0.611	**0.712**	activation’: ’tanh’, ’alpha’: 0.0001, ’early_stopping’: True, ’hidden_layer_sizes’: (150, 60, 30), ’max_iter’: 10000, ’solver’: ’adam’
	18 weeks	LR	**0.818**	0.852	**0.688**	**0.770**	C’: 10, ’max_iter’: 10000, ’penalty’: ’l2’, ’solver’: ’liblinear’
		RF	**0.766**	**0.869**	0.375	0.622	criterion’: ’gini’, ’max_depth’: 14, ’min_samples_split’: 2, ’n_estimators’: 100
		XGB	0.792	**0.902**	0.375	0.638	learning_rate’: 1.0, ’loss’: ’exponential’, ’n_estimators’: 500
		MLP	0.714	0.721	**0.688**	0.704	activation’: ’tanh’, ’alpha’: 0.01, ’early_stopping’: True, ’hidden_layer_sizes’: (100, 50, 20), ’max_iter’: 10000, ’solver’: ’adam’
Outcome	**Outcome at**	**Model**	**Accuracy**	**Specificity**	**Recall**	**AUC**	**Hyperparameters**
Usual activities	6 weeks	LR	0.761	0.797	**0.571**	**0.684**	C’: 10, ’max_iter’: 10000, ’penalty’: ’l2’, ’solver’: ’newton-cg’
		RF	**0.807**	**0.905**	0.286	0.596	criterion’: ’gini’, ’max_depth’: 18, ’min_samples_split’: 2, ’n_estimators’: 100
		XGB	**0.818**	0.892	0.429	0.660	learning_rate’: 1.0, ’loss’: ’log_loss’, ’n_estimators’: 500
		MLP	0.773	0.838	0.429	0.633	activation’: ’relu’, ’alpha’: 1e-05, ’early_stopping’: True, ’hidden_layer_sizes’: (100, 50, 20), ’max_iter’: 10000, ’solver’: ’adam’
	12 weeks	LR	**0.829**	0.861	**0.600**	**0.731**	C’: 100, ’max_iter’: 10000, ’penalty’: ’l2’, ’solver’: ’liblinear’
		RF	**0.854**	**0.958**	0.100	0.529	criterion’: ’gini’, ’max_depth’: 14, ’min_samples_split’: 2, ’n_estimators’: 100
		XGB	0.841	0.917	0.300	0.608	learning_rate’: 1.0, ’loss’: ’exponential’, ’n_estimators’: 500
		MLP	0.829	0.889	0.400	0.644	activation’: ’relu’, ’alpha’: 0.01, ’early_stopping’: True, ’hidden_layer_sizes’: (100, 50, 20), ’max_iter’: 10000, ’solver’: ’adam’
	18 weeks	LR	0.896	0.896	**0.900**	**0.898**	C’: 10, ’max_iter’: 10000, ’penalty’: ’l2’, ’solver’: ’newton-cg’
		RF	**0.909**	**0.985**	0.400	0.693	criterion’: ’gini’, ’max_depth’: 14, ’min_samples_split’: 2, ’n_estimators’: 100
		XGB	**0.909**	**0.925**	0.800	0.863	learning_rate’: 1.0, ’loss’: ’log_loss’, ’n_estimators’: 100
		MLP	**0.857**	0.851	**0.900**	**0.875**	activation’: ’relu’, ’alpha’: 0.01, ’early_stopping’: True, ’hidden_layer_sizes’: (150, 60, 30), ’max_iter’: 10000, ’solver’: ’adam’
Outcome	**Outcome at**	**Model**	**Accuracy**	**Specificity**	**Recall**	**AUC**	**Hyperparameters**
Mobility	6 weeks	LR	0.852	0.859	**0.800**	**0.829**	C’: 1, ’max_iter’: 10000, ’penalty’: ’l2’, ’solver’: ’liblinear’
		RF	**0.875**	**0.949**	0.300	0.624	criterion’: ’entropy’, ’max_depth’: 18, ’min_samples_split’: 2, ’n_estimators’: 100
		XGB	0.852	0.910	0.400	0.655	learning_rate’: 1.0, ’loss’: ’exponential’, ’n_estimators’: 500
		MLP	0.830	0.846	0.700	**0.773**	activation’: ’relu’, ’alpha’: 1e-05, ’early_stopping’: True, ’hidden_layer_sizes’: (100, 50, 20), ’max_iter’: 10000, ’solver’: ’adam’
	12 weeks	LR	0.866	0.905	**0.500**	0.703	C’: 100, ’max_iter’: 10000, ’penalty’: ’l2’, ’solver’: ’liblinear’
		RF	0.902	**0.919**	0.750	0.834	criterion’: ’gini’, ’max_depth’: 8, ’min_samples_split’: 2, ’n_estimators’: 100
		XGB	**0.890**	0.919	0.625	0.772	learning_rate’: 1.0, ’loss’: ’log_loss’, ’n_estimators’: 100
		MLP	**0.927**	**0.932**	**0.875**	**0.904**	activation’: ’relu’, ’alpha’: 1e-05, ’early_stopping’: True, ’hidden_layer_sizes’: (150, 60, 30), ’max_iter’: 10000, ’solver’: ’adam’
	18 weeks	LR	**0.896**	0.930	**0.500**	**0.715**	C’: 10, ’max_iter’: 10000, ’penalty’: ’l2’, ’solver’: ’liblinear’
		RF	**0.896**	**0.958**	0.167	0.562	criterion’: ’gini’, ’max_depth’: 8, ’min_samples_split’: 2, ’n_estimators’: 100
		XGB	**0.909**	**0.958**	0.333	0.646	learning_rate’: 0.1, ’loss’: ’exponential’, ’n_estimators’: 500
		MLP	0.883	0.915	**0.500**	0.708	activation’: ’relu’, ’alpha’: 0.0001, ’early_stopping’: True, ’hidden_layer_sizes’: (150, 60, 30), ’max_iter’: 10000, ’solver’: ’adam’

Table 6. Feature importance ranks for physical well-being changes at 18 weeks prediction models LR and RF.

Physical well-being							
Deterioration				Improvement			
Logistic regression		Random forest		Logistic regression		Random forest	
PWB_overall0	1.569433	PWB_overall0	0.133532	CMResCOPD	13.923359	PWB_overall0	0.221759
CMRheuArth	1.149526	BCBMI	0.09927	CMCarMI	13.066001	BCBMI	0.073412
CMStomInflamm	1.026601	FWB_overall0	0.081032	CMCarAngina	9.672283	EQ5DPain0	0.071416
PrimaryorMet	0.879178	SWB_overall0	0.075772	CMEndDiabetes	4.327627	EQ5DVAS0	0.064324
PreviousChemo	0.864974	EQ5DVAS0	0.066705	CMEndHyperth	4.285474	SWB_overall0	0.062971
CMPrevMal	0.79102	DiseaseSite	0.050577	CMNeuStroke	3.695441	FWB_overall0	0.062633
CMResCOPD	0.764014	DCEmployment	0.050358	CMEndHypothy	3.601606	ROL0	0.050095
CMCarMI	0.59501	ROL0	0.050047	Comorbidities	2.922004	DiseaseSite	0.045489
DiseaseSite	0.564434	PrimaryorMet	0.041654	CMCarHyperten	2.919881	ed_lev	0.038387
CMEndDiabetes	0.501928	ed_lev	0.039405	PWB_overall0	2.850354	PrimaryorMet	0.035688
CMRheuCTD	0.444377	EQ5DAnxDep0	0.034651	CMCarVenous	2.747976	DCEmployment	0.03415
FWB_overall0	0.444371	DCMarital	0.030691	CMCarArrhythm	2.534306	DCMarital	0.028505
CMEndHypothy	0.426059	EQ5DPain0	0.028751	PreviousChemo	2.414958	EQ5DUsuAct0	0.026227
EQ5DAnxDep0	0.383677	EQ5DUsuAct0	0.028591	CMRheuArth	1.93644	Comorbidities	0.025013
EQ5DPain0	0.364803	Comorbidities	0.028133	PrimaryorMet	1.54155	EQ5DAnxDep0	0.024492
CMCarVenous	0.332789	EQ5DMob0	0.021961	CMResAsthma	1.499618	EQ5DMob0	0.020482
CMResAsthma	0.289044	StudyArm	0.019965	CMPrevMal	1.319404	StudyArm	0.020222
StudyArm	0.28145	PreviousChemo	0.016012	CMSubstAlcohol	1.042471	PreviousChemo	0.013142
BCBMI	0.252698	CMCarHyperten	0.015805	DiseaseSite	0.893342	CMEndDiabetes	0.013136
CMEndHyperth	0.243022	CMResCOPD	0.012084	ed_lev	0.864195	CMCarHyperten	0.01048
CMGasPancreas	0.228956	CMResAsthma	0.011007	EQ5DVAS0	0.832183	CMEndHypothy	0.009797
ROL0	0.224995	EQ5DSelCar0	0.010529	EQ5DSelCar0	0.796896	EQ5DSelCar0	0.009597
CMStomUlcers	0.200148	CMPrevMal	0.009927	CMGasPancreas	0.74042	CMPrevMal	0.008038
CMNeuStroke	0.197854	CMEndDiabetes	0.009848	CMRheuLupus	0.703934	CMResAsthma	0.006365
CMRenEndStage	0.178679	CMCarMI	0.006526	BCBMI	0.599315	CMCarAngina	0.004794
ed_lev	0.135084	CMEndHypothy	0.006286	FWB_overall0	0.560101	CMCarMI	0.003373
Comorbidities	0.127383	CMStomInflamm	0.004816	CMNeuParkins	0.392506	CMEndHyperth	0.002924
CMCarAngina	0.123243	CMCarVenous	0.004777	CMStomInflamm	0.349838	CMCarVenous	0.002866
EQ5DVAS0	0.110928	CMCarAngina	0.002398	DCEmployment	0.189108	CMNeuParkins	0.00223
CMCarArrhythm	0.106706	CMNeuStroke	0.001957	DCMarital	0.180942	CMResCOPD	0.001576
SWB_overall0	0.102555	CMRheuArth	0.001828	EQ5DMob0	0.171249	CMCarArrhythm	0.001128
DCEmployment	0.100668	CMRheuLupus	0.001095	CMStomUlcers	0.149692	CMRheuArth	0.001108
EQ5DMob0	0.081411	CMCarArrhythm	0.000916	ROL0	0.100497	CMStomInflamm	0.001044
CMCarHeartFail	0.078057	CMRheuCTD	0.000801	SWB_overall0	0.099762	CMNeuStroke	0.000839
CMSubstAlcohol	0.076215	CMEndHyperth	0.000681	EQ5DAnxDep0	0.085732	CMGasPancreas	0.000631
DCMarital	0.07464	CMGasPancreas	0.000615	StudyArm	0.080389	CMSubstAlcohol	0.00056
EQ5DUsuAct0	0.022877	CMSubstAlcohol	0.000385	EQ5DUsuAct0	0.042851	CMRheuCTD	0.000482
CMCarHyperten	0.012993	CMRenEndStage	0.000346	EQ5DPain0	0.042741	CMRheuLupus	0.00029
EQ5DSelCar0	0.010388	CMStomUlcers	0.000227	CMRheuCTD	0.035837	CMStomUlcers	0.000284
CMRheuLupus	0.010197	CMCarHeartFail	0.00004	CMResChronBron	0.005954	CMResChronBron	0.000052
CMSubstDrugs	0	CMSubstDrugs	0	CMSubstDrugs	0	CMGasChronHep	0
CMResChronBron	0	CMGasChronHep	0	CMStomMalabsor	0	CMRheuRhPolymy	0
CMStomMalabsor	0	CMResChronBron	0	CMRheuRhPolymy	0	CMResEmphys	0
CMResEmphys	0	CMNeuMyasth	0	CMGasCirrhosis	0	CMRheuPolymyo	0
CMRheuRhPolymy	0	CMNeuParkins	0	CMRheuPolymyo	0	CMSubstDrugs	0
CMNeuMyasth	0	CMNeuMS	0	CMGasChronHep	0	CMNeuMyasth	0
CMGasChronHep	0	CMGasCirrhosis	0	CMResEmphys	0	CMNeuMS	0
CMNeuParkins	0	CMRheuRhPolymy	0	CMRenEndStage	0	CMCarHeartFail	0
CMGasCirrhosis	0	CMRheuPolymyo	0	CMCarHeartFail	0	CMGasCirrhosis	0
CMNeuMS	0	CMStomMalabsor	0	CMNeuMyasth	0	CMRenEndStage	0
CMRheuPolymyo	0	CMResEmphys	0	CMNeuMS	0	CMStomMalabsor	0

Table 7. Feature importance ranks for usual activities changes at 18 weeks prediction models LR and RF.

Usual activities							
Deterioration				Improvement			
Logistic regression		Random forest		Logistic regression		Random forest	
EQ5DUsuAct0	2.348261	BCBMI	0.143117	CMSubstAlcohol	7.569993	EQ5DUsuAct0	0.252763
CMRheuLupus	1.244129	EQ5DUsuAct0	0.129836	CMRheuArth	3.585515	ROL0	0.115769
DiseaseSite	0.898524	DiseaseSite	0.077861	EQ5DUsuAct0	3.522745	PWB_overall0	0.08537
CMPrevMal	0.865107	EQ5DVAS0	0.07398	CMCarVenous	2.355063	BCBMI	0.08323
CMResAsthma	0.771298	SWB_overall0	0.068102	CMCarArrhythm	2.165607	EQ5DVAS0	0.063433
CMCarHyperten	0.679204	PWB_overall0	0.062277	CMStomInflamm	2.008561	SWB_overall0	0.061488
CMGasPancreas	0.663488	ROL0	0.060521	CMEndHypothy	1.993451	DCEmployment	0.039528
CMEndHypothy	0.542003	DCEmployment	0.045276	CMRheuLupus	1.647547	EQ5DPain0	0.032502
BCBMI	0.506767	EQ5DAnxDep0	0.0362	CMResCOPD	1.407293	DiseaseSite	0.026328
CMRenEndStage	0.498677	DCMarital	0.03589	CMRheuCTD	1.285911	ed_lev	0.025657
EQ5DPain0	0.497696	ed_lev	0.033316	CMCarHeartFail	1.135531	DCMarital	0.025267
EQ5DSelCar0	0.47095	EQ5DPain0	0.032078	CMCarAngina	1.051474	EQ5DAnxDep0	0.023673
CMResCOPD	0.446885	Comorbidities	0.027761	CMNeuParkins	1.035357	EQ5DMob0	0.023085
CMCarVenous	0.422538	PrimaryorMet	0.024138	PWB_overall0	0.911066	Comorbidities	0.021449
CMCarMI	0.40174	StudyArm	0.022059	CMNeuStroke	0.785979	EQ5DSelCar0	0.015177
CMCarArrhythm	0.317881	EQ5DMob0	0.020994	BCBMI	0.7132	StudyArm	0.01505
Comorbidities	0.280834	CMCarHyperten	0.017985	ed_lev	0.622446	PreviousChemo	0.013022
ROL0	0.276591	PreviousChemo	0.013151	EQ5DMob0	0.60776	PrimaryorMet	0.012692
CMEndDiabetes	0.266086	EQ5DSelCar0	0.011003	CMResAsthma	0.573449	CMPrevMal	0.009341
StudyArm	0.258236	CMEndDiabetes	0.010222	EQ5DVAS0	0.556679	CMCarHyperten	0.00828
EQ5DAnxDep0	0.254403	CMPrevMal	0.009266	PreviousChemo	0.515102	CMResAsthma	0.00717
CMStomInflamm	0.24441	CMResAsthma	0.008121	ROL0	0.472313	CMResCOPD	0.005677
EQ5DMob0	0.236114	CMEndHypothy	0.006361	CMEndHyperth	0.433106	CMCarMI	0.005379
CMRheuArth	0.224756	CMSubstAlcohol	0.003922	CMPrevMal	0.405182	CMCarVenous	0.005169
CMCarAngina	0.176192	CMCarMI	0.003742	EQ5DAnxDep0	0.362858	CMEndHypothy	0.004344
EQ5DVAS0	0.164569	CMRheuArth	0.003689	StudyArm	0.336571	CMEndDiabetes	0.004095
CMRheuCTD	0.150602	CMCarVenous	0.003277	SWB_overall0	0.327688	CMCarArrhythm	0.002829
DCEmployment	0.142009	CMResCOPD	0.002494	Comorbidities	0.162286	CMCarAngina	0.00265
CMSubstAlcohol	0.136974	CMStomInflamm	0.002476	DCMarital	0.153898	CMRheuArth	0.002563
DCMarital	0.115902	CMRheuLupus	0.002339	PrimaryorMet	0.135378	CMSubstAlcohol	0.002274
CMNeuParkins	0.104832	CMGasPancreas	0.002232	CMRenEndStage	0.11648	CMStomInflamm	0.001276
CMNeuStroke	0.097995	CMCarAngina	0.001642	CMCarMI	0.109787	CMRheuLupus	0.000868
PrimaryorMet	0.096108	CMRenEndStage	0.001634	CMEndDiabetes	0.098698	CMEndHyperth	0.000723
PreviousChemo	0.082822	CMResChronBron	0.000892	CMStomUlcers	0.091996	CMRheuCTD	0.00058
SWB_overall0	0.071112	CMNeuStroke	0.000826	CMGasPancreas	0.084078	CMRenEndStage	0.000444
CMResChronBron	0.063813	CMCarArrhythm	0.000588	EQ5DSelCar0	0.076894	CMCarHeartFail	0.000305
ed_lev	0.062876	CMEndHyperth	0.00042	DCEmployment	0.063975	CMNeuParkins	0.000191
CMEndHyperth	0.058035	CMRheuCTD	0.000215	CMCarHyperten	0.04455	CMResChronBron	0.000177
PWB_overall0	0.025423	CMNeuParkins	0.000096	EQ5DPain0	0.037622	CMNeuStroke	0.000131
CMGasChronHep	0	CMCarHeartFail	0	DiseaseSite	0.027685	CMStomUlcers	0.000052
CMCarHeartFail	0	CMGasChronHep	0	CMResChronBron	0.000317	CMGasCirrhosis	0
CMResEmphys	0	CMResEmphys	0	CMSubstDrugs	0	CMResEmphys	0
CMRheuPolymyo	0	CMSubstDrugs	0	CMGasCirrhosis	0	CMSubstDrugs	0
CMRheuRhPolymy	0	CMRheuRhPolymy	0	CMGasChronHep	0	CMGasPancreas	0
CMStomUlcers	0	CMGasCirrhosis	0	CMRheuRhPolymy	0	CMRheuPolymyo	0
CMStomMalabsor	0	CMNeuMyasth	0	CMResEmphys	0	CMNeuMyasth	0
CMSubstDrugs	0	CMNeuMS	0	CMRheuPolymyo	0	CMNeuMS	0
CMNeuMS	0	CMStomMalabsor	0	CMStomMalabsor	0	CMGasChronHep	0
CMNeuMyasth	0	CMStomUlcers	0	CMNeuMyasth	0	CMStomMalabsor	0
CMGasCirrhosis	0	CMRheuPolymyo	0	CMNeuMS	0	CMRheuRhPolymy	0
CMRheuPolymyo	0	CMResEmphys	0	CMNeuMS	0	CMStomMalabsor	0

Table 8. Feature importance ranks for mobility changes at 18 weeks prediction models LR and RF.

Mobility							
Deterioration				Improvement			
Logistic regression		Random forest		Logistic regression		Random forest	
CMResChronBron	5.681567	FWB_overall0	0.1055	CMEndHyperth	5.327951	EQ5DMob0	0.318964
CMGasPancreas	4.960102	SWB_overall0	0.091013	EQ5DMob0	4.080502	BCBMI	0.078401
CMCarHeartFail	4.945931	PWB_overall0	0.077668	CMCarAngina	3.690014	FWB_overall0	0.070807
CMCarArrhythm	4.130466	EQ5DVAS0	0.073442	CMResCOPD	3.684776	DiseaseSite	0.068345
CMCarMI	3.865736	DiseaseSite	0.051927	CMCarVenous	2.94743	SWB_overall0	0.055995
CMSubstAlcohol	3.775084	DCEmployment	0.048892	CMResChronBron	2.584025	PWB_overall0	0.051819
CMEndHyperth	3.298455	ed_lev	0.040528	CMNeuParkins	2.409078	EQ5DPain0	0.048225
CMResCOPD	3.033808	EQ5DAnxDep0	0.040475	CMCarHyperten	2.35333	EQ5DVAS0	0.042132
CMRheuArth	2.903339	Comorbidities	0.034333	CMResAsthma	2.132524	DCMarital	0.038277
CMRheuLupus	2.42351	DCMarital	0.034039	PreviousChemo	2.022009	EQ5DUsuAct0	0.037502
CMCarVenous	1.366612	EQ5DUsuAct0	0.033967	CMCarMI	1.74107	DCEmployment	0.026439
CMEndDiabetes	1.287484	EQ5DPain0	0.033657	CMEndDiabetes	1.617783	Comorbidities	0.021263
CMResAsthma	1.202195	StudyArm	0.020932	CMEndHypothy	1.571741	EQ5DAnxDep0	0.015872
PreviousChemo	1.0179	PrimaryorMet	0.019977	Comorbidities	1.328023	ed_lev	0.014929
CMNeuParkins	0.761727	EQ5DMob0	0.018281	DiseaseSite	1.312387	CMCarHyperten	0.01356
BCBMI	0.651666	CMResAsthma	0.013156	SWB_overall0	1.307746	PrimaryorMet	0.012237
DiseaseSite	0.57739	PreviousChemo	0.013074	BCBMI	1.165174	PreviousChemo	0.011418
CMPrevMal	0.561657	CMCarHyperten	0.011887	CMCarArrhythm	1.071283	StudyArm	0.00947
PrimaryorMet	0.525226	CMPrevMal	0.010103	CMPrevMal	0.984691	EQ5DSelCar0	0.008962
CMStomInflamm	0.514437	EQ5DSelCar0	0.009966	CMNeuStroke	0.911514	CMPrevMal	0.008718
Comorbidities	0.489691	CMCarArrhythm	0.009579	EQ5DUsuAct0	0.856713	CMEndDiabetes	0.007119
EQ5DPain0	0.463356	CMEndDiabetes	0.009069	FWB_overall0	0.649634	CMCarMI	0.005435
EQ5DMob0	0.433626	CMCarMI	0.008925	CMRheuArth	0.530132	CMResAsthma	0.005372
CMNeuStroke	0.422049	CMRheuArth	0.008617	DCMarital	0.480517	CMEndHypothy	0.005295
EQ5DAnxDep0	0.41443	CMEndHypothy	0.008269	PrimaryorMet	0.457481	CMCarAngina	0.003963
EQ5DSelCar0	0.396671	CMResCOPD	0.003547	StudyArm	0.292456	CMEndHyperth	0.003086
EQ5DVAS0	0.283777	CMCarVenous	0.003416	CMStomInflamm	0.249998	CMCarArrhythm	0.002581
ed_lev	0.263051	CMRheuLupus	0.00247	EQ5DSelCar0	0.224756	CMCarVenous	0.002577
CMCarHyperten	0.117272	CMStomInflamm	0.002402	CMRheuLupus	0.161346	CMResCOPD	0.002568
CMEndHypothy	0.10792	CMNeuStroke	0.002397	CMSubstAlcohol	0.118481	CMRheuArth	0.002414
CMCarAngina	0.087238	CMGasPancreas	0.001793	CMGasPancreas	0.10859	CMResChronBron	0.002178
FWB_overall0	0.0783	CMCarHeartFail	0.001535	CMRenEndStage	0.1067	CMStomInflamm	0.001095
EQ5DUsuAct0	0.073808	CMEndHyperth	0.001213	EQ5DVAS0	0.095971	CMNeuParkins	0.001007
DCMarital	0.057833	CMSubstAlcohol	0.001119	EQ5DAnxDep0	0.092543	CMSubstAlcohol	0.000952
PWB_overall0	0.057653	CMCarAngina	0.000915	ed_lev	0.087568	CMNeuStroke	0.000509
StudyArm	0.048044	CMResChronBron	0.000548	EQ5DPain0	0.075561	CMRheuLupus	0.000224
SWB_overall0	0.039861	CMNeuParkins	0.000193	DCEmployment	0.074864	CMGasPancreas	0.000151
DCEmployment	0.030947	CMResEmphys	0	PWB_overall0	0.00146	CMCarHeartFail	0.00014
CMResEmphys	0	CMSubstDrugs	0	CMRheuCTD	0.000971	CMStomMalabsor	0
CMGasChronHep	0	CMRheuRhPolymy	0	CMCarHeartFail	0.000414	CMSubstDrugs	0
CMGasCirrhosis	0	CMGasChronHep	0	CMStomUlcers	0	CMGasCirrhosis	0
CMStomUlcers	0	CMRheuCTD	0	CMNeuMyasth	0	CMGasChronHep	0
CMRheuPolymyo	0	CMNeuMyasth	0	CMStomMalabsor	0	CMRheuRhPolymy	0
CMRheuRhPolymy	0	CMNeuMS	0	CMGasChronHep	0	CMRheuPolymyo	0
CMRenEndStage	0	CMRenEndStage	0	CMGasCirrhosis	0	CMRheuCTD	0
CMNeuMS	0	CMStomMalabsor	0	CMResEmphys	0	CMResEmphys	0
CMSubstDrugs	0	CMStomUlcers	0	CMRheuRhPolymy	0	CMNeuMyasth	0
CMNeuMyasth	0	CMGasCirrhosis	0	CMRheuPolymyo	0	CMNeuMS	0
CMRheuCTD	0	CMRheuPolymyo	0	CMNeuMS	0	CMRenEndStage	0
CMStomMalabsor	0			CMSubstDrugs	0	CMStomUlcers	0
CMRheuPolymyo	0	CMResEmphys	0	CMNeuMS	0	CMStomMalabsor	0

References

1. Kowal, M., Douglas, F., Jayne, D., Meads, D.: Patient choice in colorectal cancer treatment – a systematic review and narrative synthesis of attribute-based stated preference studies. Colorectal Dis. **24**(11), 1295–1307 (2022). https://onlinelibrary.wiley.com/doi/pdf/10.1111/codi.16242
2. Xuyi, W., Seow, H., Sutradhar, R.: Artificial neural networks for simultaneously predicting the risk of multiple co-occurring symptoms among patients with cancer. Cancer Med. **10**(3), 989–998 (2021). https://onlinelibrary.wiley.com/doi/pdf/10.1002/cam4.3685
3. Shehab, M., et al.: Machine learning in medical applications: a review of state-of-the-art methods. Comput. Biol. Med. **145**, 105458 (2022)
4. Kingsley, C., Patel, S.: Patient-reported outcome measures and patient-reported experience measures. BJA Educ. **17**, 137–144 (2017)
5. Sim, J.-A., et al.: The major effects of health-related quality of life on 5-year survival prediction among lung cancer survivors: applications of machine learning. Sci. Rep. **10**, 10693 (2020)
6. Chen, Y., Hosin, A.A., George, M.J., Asselbergs, F.W., Shah, A.D.: Digitaltechnology and patient and public involvement (PPI) in routine care and clinical research—a pilot study. PLoS ONE **18**, e0278260 (2023)
7. Wójcik, Z., et al.: Using machine learning to predict unplanned hospital utilization and chemotherapy management from patient-reported outcome measures. JCO Clin. Cancer Inform. **8**, e2300264 (2024)
8. Zhou, K., Bellanger, M., Le Lann, S., Robert, M., Frenel, J.-S., Campone, M.: The predictive value of patient-reported outcomes on the impact of breast cancer treatment-related quality of life. Front. Oncol. **12**, 925534 (2022)
9. Wang, Y., et al.: Predicting late symptoms of head and neck cancer treatment using LSTM and patient reported outcomes. In: Proceedings of the International Database Engineering and Applications Symposium, vol. 2021, pp. 273–279, July 2021
10. Jha, D., et al.: Ensuring trustworthy medical artificial intelligence through ethical and philosophical principles, September 2023. arXiv:2304.11530 [cs]
11. Absolom, K., et al.: Phase III randomized controlled trial of eRAPID: eHealth intervention during chemotherapy. J. Clin. Oncol. **39**, 734–747 (2021).
12. Dolan, P.: Modeling valuations for EuroQol health states. Med. Care **35**, 1095–1108 (1997)
13. Cella, D.F., et al.: The functional assessment of cancer therapy scale: development and validation of the general measure. J. Clin. Oncol. **11**, 570–579 (1993)
14. Aaronson, N.K., et al.: The European organization for research and treatment of cancer QLQ-C30: a quality-of-life instrument for use in international clinical trials in oncology. J. Natl. Cancer Inst. **85**, 365–376 (1993)
15. King, M.T., et al.: Meta-analysis provides evidence-based effect sizes for a cancer-specific quality-of-life questionnaire, the FACT-G. J. Clin. Epidemiol. **63**, 270–281 (2010)
16. Musoro, J.Z., et al.: Minimally important differences for interpreting EORTC QLQC30 scores in patients with advanced breast cancer. JNCI Cancer Spectr. **3**, pkz037 (2019)
17. Aljrees, T.: Improving prediction of cervical cancer using KNN imputer and multimodel ensemble learning. PLoS ONE **19**, e0295632 (2024)
18. Shafique, R., et al.: Breast cancer prediction using fine needle aspiration features and upsampling with supervised machine learning. Cancers **15**, 681 (2023)
19. Christodoulou, E., Ma, J., Collins, G.S., Steyerberg, E.W., Verbakel, J.Y., Van Calster, B.: A systematic review shows no performance benefit of machine learning over logistic regression for clinical prediction models. J. Clin. Epidemiol. **110**, 12–22 (2019)

Predicting Blood Glucose Levels with LMU Recurrent Neural Networks: A Novel Computational Model

Ladislav Floriš and Daniel Vašata(✉)

Faculty of Information Technology, Czech Technical University in Prague,
Thákurova 9, Prague, Czech Republic
{florilad,daniel.vasata}@fit.cvut.cz

Abstract. Type 1 diabetes disrupts normal blood glucose regulation due to the destruction of insulin-producing cells, necessitating insulin therapy through injections or insulin pumps. Consumer devices can forecast blood glucose levels by leveraging data from blood glucose sensors and other sources. Such predictions are valuable for informing patients about their blood glucose trajectory and supporting various downstream applications. Numerous machine-learning models have been explored for blood glucose prediction.

This study introduces a novel application of Legendre Memory Units for blood glucose prediction. Employing a multivariate time series, predictions are made with 30-minute and 60-minute horizons. The proposed model is comparable with state-of-the-art models on the OhioT1DM dataset, encompassing eight weeks of data from 12 distinct patients.

Keywords: type 1 diabetes · blood glucose prediction · time series forecasting · Legendre Memory Units

1 Introduction

Type 1 diabetes (T1D) is an autoimmune disorder causing abnormal blood glucose (BG) levels due to the body's incapability to produce insulin. The current approach to treatment relies heavily on the patient's self-management, which means actively tracking glucose levels, injecting and dosing insulin, and managing diet and physical activity.

The problem of blood glucose prediction for patients with type 1 diabetes is essential for developing tools that help patients make better decisions about treatment, as well as systems for automatic insulin delivery. These systems include BG alarms that can notify the patient about an upcoming high or low glucose event, and closed-loop systems, which are capable of using the sensor glucose readings, and possibly other data sources to calculate insulin dosage and send a signal to an insulin pump to administer insulin.

Many state-of-the-art models for BG prediction utilize Long Short-Term Memory units (LSTM) [1,8,11]. This work presents a novel application of a

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 117–127, 2024.
https://doi.org/10.1007/978-3-031-66538-7_13

model based on Legendre Memory Units (LMU) [13] for BG prediction. As will be shown, on the 2018 edition of the OhioT1DM dataset [7], our model outperforms current models in both 30-minute and 60-minute prediction horizons.

The model processes past blood glucose levels, glucose levels from finger blood drop samples, insulin intake, and carbohydrate consumption in the form of a multivariate time series and predicts future blood glucose levels. Formally, as the input, the model takes n successive previous measurements of feature vectors in past n time steps $t_1, t_2, \ldots, t_n$ from interval $[t-(n-1)\Delta t, t]$, where $t_i = t-\Delta t(n-i)$, as an input. Hence, extending n or Δt brings a longer historical context to the model. Each component of a feature vector $\boldsymbol{x}_s = (x_{s;1}, x_{s;2}, \ldots, x_{s;p})$ is associated with one of the p features (BG, fingerstick, insulin, carbohydrates). The output is the prediction $\hat{G}_{t+\mathrm{PH}}$ of the BG level $G_{t+\mathrm{PH}}$ at time $t + \mathrm{PH}$, where PH is the prediction horizon indicating how long into the future the predictions are being made. In our case, the model, represented by a function $f_{\Delta}(\boldsymbol{x}_{t_1}, \ldots, \boldsymbol{x}_{t_n})$, analogously to [15] predicts the change $G_{t+\mathrm{PH}} - G_t$.

It is expected that with a longer PH, the prediction performance will decrease. It takes time for BG to change, and as PH increases, the BG has further opportunities to develop, leading to a wider range of possible BG values.

2 Dataset

This paper uses one of the most standard datasets - the OhioT1DM dataset[1] [7]. It has two versions, 2018 and 2020. We evaluate the developed models on both versions. The dataset comprises data spanning eight weeks from 12 individuals with type-1 diabetes. Patients logged daily events through a smartphone application and a fitness band. This dataset contains continuous glucose monitoring (CGM) blood glucose readings taken every 5 min, alongside self-monitored blood glucose levels via fingerstick, insulin administrations (both bolus and basal doses), and self-reported instances of meal consumption with carbohydrate estimates, physical activity, sleep, work, stress, and illness occurrences. Additionally, the dataset features 5-minute summaries of physical activity metrics, including step count and heart rate, as well as physiological and environmental measures like galvanic skin response (GSR), skin temperature, and air temperature.

3 Related Work

Various models were previously used for the BG prediction task. The most significant ones are support-vector regression [4], statistical approaches like ARIMA [14], and neural networks discussed primarily in the following. In this section, we provide a selection of approaches that use the same OhioT1DM dataset.

[1] Available at http://smarthealth.cs.ohio.edu/OhioT1DM-dataset.html.

3.1 Performance on OhioT1DM, 2018 Edition

This section introduces works that utilized the 2018 edition of the OhioT1DM dataset. Chen et al. in [1] proposed a Dilated Recurrent Neural Network for a 30-minute BG prediction task, utilizing the 1-hour input containing CGMs, bolus insulin doses, and carbohydrate intake, achieving a mean root mean squared error (mean RMSE) of 19.04. Performance was evaluated on all test data points.

Martinsson et al. in [8] implemented a LSTM based neural network that utilized the past 30 min of BG history to make BG prediction 30 and 60 min in the future, achieving a mean RMSE of 20.1 and 33.2, respectively. Performance was evaluated only on test points where at least 12 previous consecutive measurements were available.

Rabby et al. in [11] implemented a stacked LSTM neural network, which utilized the past 2 h of BG values, carbohydrate intake from the meal, insulin dose as a bolus, and 5-min aggregation of step count from the fitness band. The carbohydrate and insulin features were transformed into continuous variables instead of using the raw discrete samples. They further experimented with Kalman smoothing of the input and target BG time series. They report RMSE for both 30 and 60-minute PH. The RMSE achieved for raw BG input and output was 18.57 and 30.32, respectively. The best RMSE for the Kalman smoothed BG was 5.89 and 17.24, respectively. The paper mentions that: "It is possible to evaluate the model at a certain point in time if there are at least 24 prior data points available (prior data points for 2 h or 120 min)", which suggests that the performance was evaluated only on test points where at least 24 previous consecutive measurements were available, but it is not stated explicitly.

3.2 Performance on OhioT1DM, 2020 Edition

This section introduces works utilizing the recent 2020 edition of the OhioT1DM dataset. The reported performance is always evaluated on all provided test data points of all test patients from the 2020 edition of OhioT1DM. Some works also utilized the 2018 edition and/or other datasets for pre-training.

Zhu et al. in [15] proposed Generative Adversarial Networks (GAN) for BG prediction, utilizing 1.5 h of historical data, which contained BG, carbohydrate intake, and bolus insulin. The first half of the cohort from the 2018 OhioT1DM edition was used for model pre-training. The achieved RMSE was 18.34 and 32.21 for 30-minute and 1-hour PH, respectively.

Rubin-Falcone et al. in [12] developed an N-BEATS model, utilizing BG, finger stick glucose, bolus values, carbohydrate inputs, sine and cosine of time, and missingness indicators for BG values. They've built an ensemble of models, each using a different length of input, and used the median as the ensemble prediction. The models were pre-trained on the 2018 version of the OhioT1DM dataset as well as the Tidepool dataset [9] and then fine-tuned per patient. The achieved mean RMSE was 18.22 and 31.66 for 30-minute and 1-hour PH, respectively. The authors further mention that without pre-training on OhioT1DM 2018 and Tidepool, the 30-minute PH performance was 18.87.

Daniels et al. in [3] proposed a multi-task learning approach by training a convolutional RNN (CRNN) with subject-specific layers. The model was trained on 2-hour-long inputs with BG, insulin bolus, carbohydrate intake, and reported exercise. The achieved RMSE was 19.79 and 33.73 for 30-minute and 1-hour PH, respectively.

4 Preprocessing and Feature Engineering

The OhioT1DM dataset contains two files, train and test, for each patient. The training dataset is loaded and further split into training and validation sets, with the first 80% of the data points used for training the models and the last 20% for validation. Models evaluated on the 2020 dataset use both train and test portions of the 2018 dataset for training and validation.

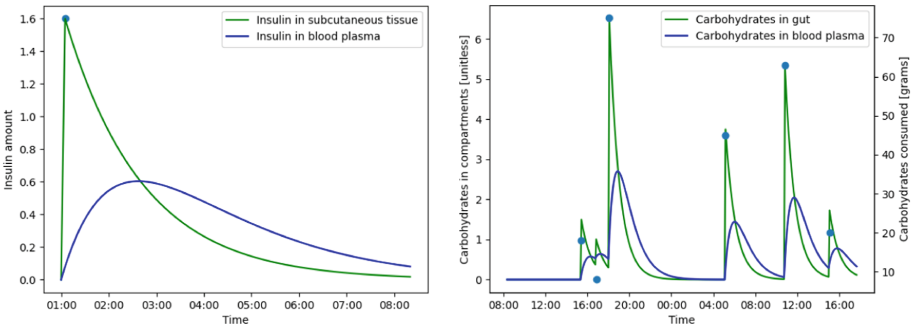
Glucose readings are expected every 5 min in the BG sensors used in the OhioT1DM study. Occasionally, gaps occur in the sensor's BG readings. This may happen due to loss of signal, malfunction, or patient not wearing the sensor. Both train and test datasets are resampled to a 5-minute sampling frequency. The missing values for carbohydrates, insulin, and fingerstick are simply set to 0. Missing BG values are first marked missing and later interpolated/extrapolated when creating the input windows (vectors) for the models (see Sect. 4.3).

The length of the input was chosen to be 30 min (6 samples). In the conducted experiments, longer input windows did not lead to better performance. Surprisingly, sometimes, they produced worse performance of the models. It is suspected that the feature engineering applied to insulin and carbohydrates might have helped with decreasing the input length needed (as discussed further). This assumption is made because the transformed carbohydrate and insulin features effectively stretch out the influence of insulin and carbohydrates in time. E.g., a model with a 30-minute input window may still see the effects of insulin taken more than 8 h ago.

4.1 Insulin Feature Engineering

Insulin models are used to bring more information to the BG prediction model about insulin properties. Specifically, instead of giving the model only raw insulin doses in time, a rate of insulin release and/or insulin concentration is provided. The rate of insulin release indicates how active the injected insulin is at a particular time. Insulin will first build up its activity and then gradually lower it.

The implementation of the Hovorka insulin model [5] is used to transform discrete bolus insulin samples into a continuous feature. The implementation is inspired by the work of Price [10]. It is a two-compartmental insulin model that provides an estimate of the amount of insulin in subcutaneous tissue and blood plasma. The parameter of the model is the time-to-max absorption of subcutaneously injected short-acting insulin, and this value was set to 100 min for all patients. Numerous values were explored for the time-to-max absorption, and it was seen that the models performed similarly well with values as low as 60 and as high as 180. Figure 1 (a) shows an example of this insulin transformation. The insulin in the subcutaneous tissue feature was used to train the models.



(a) Insulin

(b) Carbohydrate

Fig. 1. Example of transforming discrete insulin (a) and carbohydrate (b) samples into a continuous feature.

4.2 Carbohydrates Feature Engineering

Similarly to insulin, discrete carbohydrate values are also transformed into a continuous feature using an implementation from [5] of a two-compartmental meal model. The model estimates the amount of glucose available in the gut and blood plasma. A parameter of the model is the time of maximum glucose rate of appearance, which was set to 60 min for all patients, which is the same value as the one used by Rabby et al. [11], although their carbohydrates model implementation is different. Figure 1 (b) shows an example of this transformation. The carbohydrates in the gut feature were used to train the models.

4.3 Handling of Missing Blood Glucose Values

Missing BG values are first filled with the fingerstick glucose if it is available. Then, in the training and validation set, missing BG values are linearly interpolated if there are less than 7 continuous missing samples (30 min). The interpolation is done on the whole sets, meaning that both inputs and targets of the models during training and validation may be interpolated.

Interpolation is not applied to the test dataset as a whole since this could introduce data leakage. Here, for each input and target, the following decision-making process is applied: (1) if all input BG values are missing, the input-target pair is dropped, (2) if the target BG value is missing, the input-target pair is dropped; (3) else, that is, when the target value exists, and input BG has at least 1 value, the rest of the missing samples in the input are interpolated and forward and backward filled from the existing BG values in the input.

5 Model

Our proposed model is a neural network f_Δ consisting of one hidden layer of recurrent Legendre memory unit (LMU) cells followed by a single output neuron that predicts the change $G_{t+\mathrm{PH}} - G_t$ of the BG values at t and $t + \mathrm{PH}$. The final BG prediction is then given by $\hat{G}_{t+\mathrm{PH}} = f_\Delta(\boldsymbol{x}_{t_1}, \dots, \boldsymbol{x}_{t_n}) + G_t$.

5.1 Legendre Memory Units

Legendre Memory Units (LMUs), introduced in [13], are a relatively less known type of recurrent neural network (RNN) architecture designed to process time-series data efficiently. They leverage the mathematical properties of Legendre polynomials to create a fixed-size memory cell, enabling them to achieve high precision in capturing and representing temporal information over sequences with long-range dependencies.

Having the input $\boldsymbol{x}_s$ at time s, memory $\boldsymbol{m}_{s-1}$ and hidden state $\boldsymbol{h}_{s-1}$ at previous time $s-1$, the LMU cell first computes the memory input as $u_s = \boldsymbol{e}_x^T \boldsymbol{x}_s + \boldsymbol{e}_h^T \boldsymbol{h}_{s-1} + \boldsymbol{e}_m^T \boldsymbol{m}_{s-1}$ where $\boldsymbol{e}_x, \boldsymbol{e}_h, \boldsymbol{e}_m$ are trainable parameters. Then it updates the memory vector as $\boldsymbol{m}_s = \mathbf{A}\boldsymbol{m}_{s-1} + \mathbf{B}u_s$, where $\mathbf{A}, \mathbf{B}$ are trainable parameters. Finally, it uses a hidden unit g with its own trainable parameters to produce the output being also a new hidden state as $\boldsymbol{h}_s = g(\boldsymbol{x}_s, \boldsymbol{m}_s, \boldsymbol{h}_{s-1})$.

Table 1. Optimal hyper-parameters for LMU models tuned on 2018 and 2020 OhioT1DM cohorts and 30-minute and 60-minute PH

Cohort	PH [min]	order	HM	MM	IH	units	dropout	recurrent dropout
2018	30	52	True	False	False	72	0.1	0
2018	60	54	False	True	False	144	0.1	0
2020	30	60	True	False	False	72	0.1	0.2
2020	60	64	False	False	True	156	0.1	0.1

5.2 Hyper-parameter Tuning

LMUs have several hyper-parameters that can be set: hidden-to-memory connection (HM), memory-to-memory learnable connection (MM), input-to-hidden connection (IH), memory dimension, number and type of hidden units, dropout, and recurrent dropout rate, and the order, which is the number of degrees in the transfer function of the linear time-invariant system used to represent the sliding window of history. The type of hidden units in the LMU cells was chosen to be LSTM as it was experimentally proven to achieve the best performance. The memory dimension was set to 4 as this respects the dimensionality of the input, the rest of the hyper-parameters were tuned. Furthermore, the input length was manually tuned, and a 30-minute long input history led to the best performance.

The other hyper-parameters of the model were tuned using Hyperband [6]. The tuning was performed separately for the individual dataset editions and PHs. The final optimal values of hyper-parameters can be seen in Table 1.

5.3 Model Implementation and Training

The models for the 2018 OhioT1DM edition were trained on the data from all patients from the 2018 cohort. The models for the 2020 edition were trained on both train and test portions of the 2018 edition and on the training datasets of all patients from the 2020 cohort. All models utilized the same input features: BG, fingerstick, insulin in subcutaneous tissue, and carbohydrates in the gut. The input length of 30 min (6 samples) also remained the same for all models. The evaluated PH was 30 min and 1 h.

All models were trained and evaluated 5 times. All of them were trained using Adam optimizer with a learning rate set to 10^{-3}, maximum of 300 epochs and early stopping with the patience of 35, meaning the model training was stopped if validation loss did not improve in 35 epochs. The loss function used was mean squared error. Furthermore, a learning rate reducer was used, which reduces the learning rate during the training by a factor of 10^{-1} when validation loss stagnates for 10 epochs. Model checkpoints were implemented to preserve the model with the best validation loss in each training.

Source codes of all performed experiments and further details about the model architecture can be found at https://github.com/MrBlueHere/Predicting-Blood-Glucose-Levels-with-LMU-RNNs.

6 Results and Discussion

Each model is evaluated by calculating the RMSE for each patient and computing the mean RMSE as the mean of RMSEs of all patients. It should be noted that this leads to each patient's RMSE having the same weight in the final mean RMSE, even though the number of test samples is different between patients.

Table 2. Test evaluation of the best LMU model on the 2018 OhioT1DM edition.

Patient ID	RMSE (30-min PH)	RMSE (60-min PH)
559	17.63 ± 0.04	30.81 ± 0.14
563	17.54 ± 0.06	29.24 ± 0.14
570	15.23 ± 0.06	27.15 ± 0.12
575	21.28 ± 0.07	33.59 ± 0.18
588	16.83 ± 0.05	28.24 ± 0.14
591	20.51 ± 0.04	32.92 ± 0.19
Mean RMSE	18.17 ± 0.02	30.33 ± 0.08

Table 3. Test evaluation of the best LMU model on the 2020 OhioT1DM edition.

Patient ID	RMSE (30-min PH)	RMSE (60-min PH)
540	20.50 ± 0.05	37.70 ± 0.14
544	16.51 ± 0.07	28.72 ± 0.17
552	15.72 ± 0.06	28.98 ± 0.11
567	20.41 ± 0.06	36.13 ± 0.15
584	21.92 ± 0.07	35.90 ± 0.16
596	16.32 ± 0.06	28.00 ± 0.13
Mean RMSE	18.56 ± 0.03	32.57 ± 0.09

Table 4. Test Clarke error grid distribution (in %) on the 2020 OhioT1DM edition.

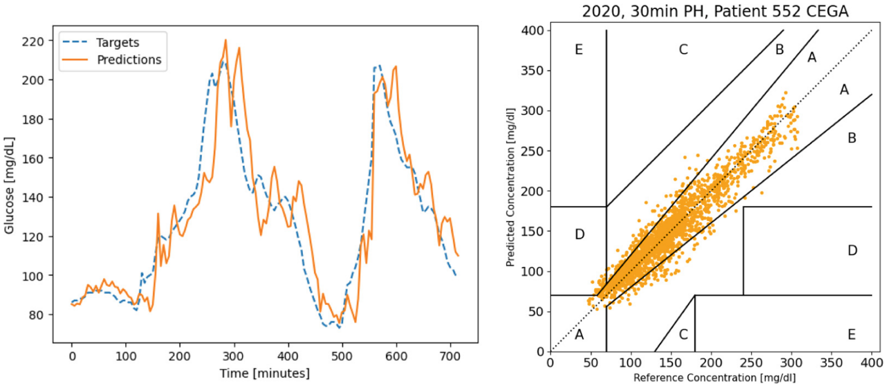
	Zone for 30-min PH					Zone for 60-min PH				
Patient ID	A	B	C	D	E	A	B	C	D	E
540	85.25	12.43	0	2.33	0	60.06	34.02	0.10	5.82	0
544	92.62	6.90	0	0.48	0	74.02	24.37	0	1.61	0
552	90.48	8.32	0	1.20	0	66.29	30.23	0.04	3.40	0.04
567	86.80	10.94	0	2.26	0	59.57	31.65	0.30	8.48	0
584	87.33	11.64	0.08	0.95	0	69.32	28.16	0.50	2.02	0
596	90.82	7.19	0	1.99	0	73.32	23.71	0.04	2.94	0

Table 5. Mean RMSE comparison on the 2018 OhioT1DM edition.

Model	Mean RMSE (30-min PH)	Mean RMSE (60-min PH)
LSTM [8]	20.1	33.2
Dilated RNN [1]	19.04	not-applicable
Stacked LSTM [11]	18.57	30.32
LMU (ours)	18.17	30.33

Table 6. Mean RMSE comparison on the 2020 OhioT1DM edition.

Model	Mean RMSE (30-min PH)	Mean RMSE (60-min PH)
GAN [15]	18.34	32.21
N-BEATS [12]	18.22	31.66
CRNN [3]	19.79	33.73
LMU (ours)	18.56	32.57



(a) Predictions

(b) Clarke error grid

Fig. 2. Example of BG predictions (a) and the Clarke error grid plot (b) for 30-minute PH and the patient with ID 552 on 2020 OhioT1DM edition.

Tables 2 and 3 show the performance of the evaluated LMU models. Figure 2 (a) shows an example of predictions made by the best LMU model. Moreover, Tables 5 and 6 show the comparison of the best LMU models with other works mentioned in the related work, evaluated on the 2018 and 2020 editions of OhioT1DM, respectively.

It can be seen that the performance on the 2018 edition of the dataset is the best in the case of 30-min PH and comparable to the best in the case of the 60-min PH. One should also note that for the Stacked LSTM introduced by Rabby et al. in [11], the authors skipped testing inputs with any missing values of BG. If we do it as well, the results are further improved to 18.03 mean RMSE for 30-min PH and 30.18 mean RMSE for 60-min PH.

The performance of our model on the 2020 edition of OhioT1DM is slightly worse than the state-of-the-art model N-BEATS [12]. However, as was mentioned in Sect. 3.2, they use more datasets for training which has a positive influence.

In addition to RMSE, we evaluated the Clarke error grid [2] as a semi-quantitative tool to obtain clinical relevance. An example is provided in Fig. 2 (b) , and the values for all patients on the 2020 edition of OhioT1DM are reported in Table 4. The results confirm the promising performance.

7 Conclusion

This work presents a novel application of Legendre Memory Units on the task of blood glucose prediction. We focus on both 30-minute and 60-minute prediction horizons. Based on the experiments conducted, LMUs were proven to reach and, on smaller datasets (2018 edition of OhioT1DM), even outperform the state-of-the-art models. Further, the LMUs can achieve such performance while utilizing significantly fewer parameters than competing models. This would be especially important if the network should be deployed on an embedded device requiring an energy-efficient implementation. The success of the proposed model is possibly due to the ability of LMUs to capture long-range dependencies. Future work should focus more on the observation of this phenomenon and also on more sophisticated preprocessing, which seems to have a substantial impact.

Acknowledgments. This work was supported by the Student Summer Research Program 2023 of FIT CTU in Prague, Czech Republic.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Chen, J., Li, K., Herrero, P., Zhu, T., Georgiou, P.: Dilated recurrent neural network for short-time prediction of glucose concentration. In: KHD@ IJCAI, pp. 69–73 (2018)
2. Clarke, W.L., Cox, D., Gonder-Frederick, L.A., Carter, W., Pohl, S.L.: Evaluating clinical accuracy of systems for self-monitoring of blood glucose. Diabetes Care **10**(5), 622–628 (1987). https://doi.org/10.2337/diacare.10.5.622
3. Daniels, J., Herrero, P., Georgiou, P.: Personalised glucose prediction via deep multitask networks. KDH@ ECAI **20**, 110–114 (2020)
4. Georga, E.I., et al.: Multivariate prediction of subcutaneous glucose concentration in type 1 diabetes patients based on support vector regression. IEEE J. Biomed. Health Inform. **17**(1), 71–81 (2013). https://doi.org/10.1109/TITB.2012.2219876
5. Hovorka, R., et al.: Nonlinear model predictive control of glucose concentration in subjects with type 1 diabetes. Physiol. Meas. **25**(4), 905 (2004)
6. Li, L., Jamieson, K., DeSalvo, G., Rostamizadeh, A., Talwalkar, A.: Hyperband: a novel bandit-based approach to hyperparameter optimization. J. Mach. Learn. Res. **18**(185), 1–52 (2018). http://jmlr.org/papers/v18/16-558.html
7. Marling, C., Bunescu, R.: The OhioT1DM dataset for blood glucose level prediction: update 2020. In: CEUR Workshop Proceedings, vol. 2675, pp. 71–74 (2020). https://ceur-ws.org/Vol-2675/paper11.pdf
8. Martinsson, J., Schliep, A., Eliasson, B., Meijner, C., Persson, S., Mogren, O.: Automatic blood glucose prediction with confidence using recurrent neural networks. In: Proceedings of the 3rd International Workshop on Knowledge Discovery in Healthcare Data, KDH@IJCAI-ECAI 2018, pp. 64–68 (2018)
9. Neinstein, A., et al.: A case study in open source innovation: developing the tidepool platform for interoperability in type 1 diabetes management. J. Am. Med. Inform. Assoc. **23**(2), 324–332 (2016)
10. Price, T.: Working repository for master thesis. https://github.com/ThonyPrice/Master_Thesis/ (2019). Accessed 20 Mar 2024
11. Rabby, M.F., Tu, Y., Hossen, M.I., Lee, I., Maida, A.S., Hei, X.: Stacked LSTM based deep recurrent neural network with Kalman smoothing for blood glucose prediction. BMC Med. Inform. Decis. Mak. **21**, 1–15 (2021)
12. Rubin-Falcone, H., Fox, I., Wiens, J.: Deep residual time-series forecasting: application to blood glucose prediction. KDH@ ECAI **20**, 105–109 (2020)

13. Voelker, A., Kajić, I., Eliasmith, C.: Legendre memory units: continuous-time representation in recurrent neural networks. Adv. Neural Inform. Process. Syst. **32** (2019)
14. Yang, J., Li, L., Shi, Y., Xie, X.: An ARIMA model with adaptive orders for predicting blood glucose concentrations and hypoglycemia. IEEE J. Biomed. Health Inform. **23**(3), 1251–1260 (2019). https://doi.org/10.1109/JBHI.2018.2840690
15. Zhu, T., Yao, X., Li, K., Herrero, P., Georgiou, P.: Blood glucose prediction for type 1 diabetes using generative adversarial networks. In: CEUR Workshop Proceedings, vol. 2675, pp. 90–94 (2020). https://ceur-ws.org/Vol-2675/paper15.pdf

Prediction Modelling and Data Quality Assessment for Nursing Scale in a Big Hospital: A Proposal to Save Resources and Improve Data Quality

Chiara Dachena[1], Roberto Gatta[2], Mariachiara Savino[3](✉), Stefania Orini[2,5], Nicola Acampora[4], M. Letizia Serra[4], Stefano Patarnello[1], Christian Barillaro[4], and Carlotta Masciocchi[1]

[1] Real World Data Facility, Gemelli Generator, Fondazione Policlinico Universitario A. Gemelli IRCCS, Rome, Italy
[2] Department of Clinical and Experimental Sciences, University of Brescia, Brescia, Italy
[3] Diagnostica per Immagini, Radioterapia Oncologica ed Ematologia, Università Cattolica del Sacro Cuore, Rome, Italy
mariachiarasavino@unicatt.it
[4] Fondazione Policlinico Universitario Agostino Gemelli IRCCS, Rome, Italy
[5] Alzheimer's Unit - Memory Clinic, IRCCS Istituto Centro San Giovanni di dio Fatebenefratelli, Brescia, Italy

Abstract. Nursing scales play an important role in the evaluation of patients' clinical and social frailty. Filling out correctly the scales, allows the early identification of patients at risk of prolonged hospitalization or difficult discharge, and enables an estimation of care complexity during hospitalization. Given the high predictive value of these scales, it is important that the measurements are reported precisely. In this paper, we provide a general methodology to estimate the quality of data related to nursing scales and we introduce an approach to infer a certain scale based on other ones that share common fields in their calculation. The former is for measuring the reliability of the scoring values, and the latter is to reduce data entry time and costs. Our experimental setting focuses on two scales: Blaylock Risk Assessment Screening Score Index (Brass scale), that evaluates the risk of difficult discharge, and Care Dependency Index (IDA scale), that evaluates the degree of care complexity. These two scales have several fields which often provide similar or correlated information about the patient's clinical condition. Preliminary results show the possibility to reduce oversights in filling out the scales use an automatic evaluation metric, which allows reproducing better the clinical condition of the patient. Moreover, the opportunity to predict value of one scale using the data of the other one allows the nurse to reduce the time for completing and to focus more on patient care.

Keywords: Data Quality Assessment · Real World Data · Nursing Scale · Prediction Model

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 128–137, 2024.
https://doi.org/10.1007/978-3-031-66538-7_14

1 Introduction

Clinical assessment is assuming an increasingly crucial role in evaluating and monitoring patients throughout their care pathway, enabling the identification of care needs, diagnostic pathways, and appropriate settings, while also providing prognostic value. However, the increase in bureaucratic workload and the imperative to "quantify clinical data", both during initial assessments and throughout hospitalization and discharge, are leading to a significant rise in the number of scales demanded from healthcare providers. This approach risks to compromise the quality of acquired data, reducing the time dedicated to patient care and interaction, and increasing the risk of caregiver burnout. In some cases, different tools require similar or overlapping information, and the clinical benefit may not be immediate or readily tangible. Therefore, within the daily clinical activities of healthcare providers, these tools may be perceived as an unnecessary burden, leading to a diminished attention and accuracy in completing the scales, as well as rendering them ineffective for their intended purposes.

The use of Electronic Health Records (EHRs) has increased data availability without requiring subsequent input. However, compared to prospectively collected data for pharmacological research, which undergo subsequent reviews and checks, data collected retroactively from EHRs exhibit undoubtedly inferior quality due to collection circumstances (a high number of different operators, sometimes inadequately trained, working conditions with frequent interruptions, sometimes a lack of awareness of their utility within daily activities). For this reason, several works about the data quality control in real-world data from EHR are proposed [4, 5, 8, 9].

Filling out the scales precisely requires time from nurses that could be dedicated to patient care. For this reason, scales are filled out precipitously, with a higher chance of running into oversights. Our objective is to evaluate two scales utilized in our hospital by nursing staff upon patient admission and according to a well-defined timing. These two scales were developed for different purposes and aim to address two distinct aspects of care needs. Along with others, such tools are part of the nursing flowsheet defined in the hospital protocol and are used by nurses to assess the clinical and physical condition of patients in the early days of hospitalization and whenever the patient's clinical condition changes. Obviously, some fields of the different scales have overlaps.

The first scale is the Index of Dependency Assessment (IDA) [2], which determines the necessary level of care during hospitalization. Its primary goal is to define the patient's optimal care setting and needs during hospitalization, considering his or her clinical severity and prognostic stratification. The use of this scale could reduce potential complications and contain costs without exposing patients to additional risks. IDA is based on a series of dependency variables, assessed through professional involvement, with a scoring system identifying patients with high care complexity (score from 7 to 11), medium complexity (score from 12 to 19), and low complexity (score from 20 to 28).

The second scale is the Blaylock Risk Assessment Screening Score (Brass) [1], which identifies patients at risk of prolonged hospital stays or challenging home discharge: bed occupancy due to one of the previous reasons, in a structural deficit situation, could cause overcrowding in emergency departments. The Brass scale comprises 20 elements, including age, housing situation, functional status, cognitive and behavioral patterns, mobility, sensory deficits, and previous hospitalizations. Depending on the final score, patients may with prolonged hospital stays may be considered with one of the three following risk levels for patients: low risk, medium risk, and high risk, with the highest class activating the care continuity unit. This risk screening scale can be used from the beginning of hospitalization to identify patients in need of a discharge plan. In our facility, such assessment directly impacts the activation of standardized company procedures. A Brass score equal to or greater than 20, indicating a population of patients at high risk of prolonged hospitalization, needs an evaluation by the Central Care Continuity (CCA) unit [3, 7]. A patient with a medium risk, i.e. with a Brass score between 11 and 19, requires an additional evaluation from the CCA unit in case any additional need emerges from the clinical evaluation.

To avoid the risk of a non-correct evaluation of the patient clinical condition, the concept of developing a metric to assess the reliability of each scale measurement is being considered. Moreover, the high overlap among the two scales, suggests a possibility to identify a score that quantifies the correct overlap between two measurements performed in a short time period. Analyzing together each measurement and the score that evaluate the overlap among scales, it is possible to define a subgroup of reliability scores to use in a predictive model.

The aim of this predictive model is to obtain the final value of one scale starting from the data of the other one.

2 Materials and Methods

Adult patients (over 18 years old) admitted from January 1st 2020 to December 31st 2022 at Fondazione Policlinico Universitario A. Gemelli IRCCS in Rome (Italy), who transited at least once to a medical area department during their hospitalization were included in the study. Clinical data were automatically extracted from the hospital's data warehouse and processed by the Gemelli Generator Real World data (G2 RWD) facility.

For these patients, all measurements of Brass and IDA scales collected during the hospitalization are analyzed. The selected cohort includes 16.534 patients and 21.544 hospitalization, 39.544 Brass scales (16.287 patients (98%) with at least one Brass measurement and 20.877 hospitalization (97%) with at least one Brass measurement) and 184.570 IDA scales (15.408 patients (93%) with at least one IDA measurement and 19.674 hospitalization (91%) with at least one IDA measurement). Patients with at least both one Brass and IDA measurements represent the final cohort, with 15.269 patients and 19.411 hospitalizations. This final patient cohort covers more than 50 inpatient wards, and scales were performed by more than 900 different nurses.

For this analysis, the first step is to calculate of each scale the metric described in the following Sect. 2.1. This pre-processing step was performed using R [6]. Following the pipeline, a good reliability is expressed with a low metric value. Secondary, all couples of Brass and IDA scale for each patients are considered. As illustrated in the following Sect. 2.2, scales within a maximum time interval of 2 h are used for the overlap analysis. Analyzing the data, measurements are close in time as to mean that are often executed one after the other by the nurses.

To identify the reliable measurements, we decide to tolerate a 20% of error on the overlap between the two scales and on the metric calculated for each scale. For this reason, a good overlap is considered if 5 out of 6 conditions reported in the following Sections are verified. Brass and IDA scales with a metric value less or equal to 1 and 1.3, respectively, are taken into account in the analysis. Finally, of the remaining scale measurements, only those with a maximum time of 2 h are analyzed. 18.630 couples of Brass and IDA scales are effectively used to analyze a predictive model.

2.1 Data Quality Assessment: Proposed Metric

Within both scales, there are fields that could be evaluated automatically from the patient hospitalization information, i.e. age in Brass and origin in IDA. The correct compilation of these two fields is easily obtained by cross-referring information about the hospitalization. At the same time, there are some fields in both scales that could influence the other ones, such as comatose state in the cognition field affects the mobility field, which will only be non-ambulatory. These simple examples are the basis for the construction of the single scale assessment metrics. For each scale, incompatible clinical condition extracted from the choices of the fields are highlighted, and a final score for the measurement is assigned. The score is evaluated based on the severity of the association. The more severe the discordance between fields, the greater will be the score.

Brass Metric. As described above, the Brass scale is composed of 20 items and the total value may range from 0 to 40, and it is obtained by adding the value from each item. The proposed pipeline is showed in Fig. 1:

- If the *age* is wrong, a value equal to 1 is assigned to the metric. This is the highest score since the value is easily obtainable by the nurse.
- If the *cognition field* is set to "comatose", the *mobility field* will only be "non-ambulatory". Different choices in the *mobility field* are evaluated with a score equal to 0.9.
- At the same time, if the *cognition field* is set to "comatose", the *functional field* cannot be filled with the choice "autonomous". If such situation occurs, a value equal to 0.9 is added to the metric.

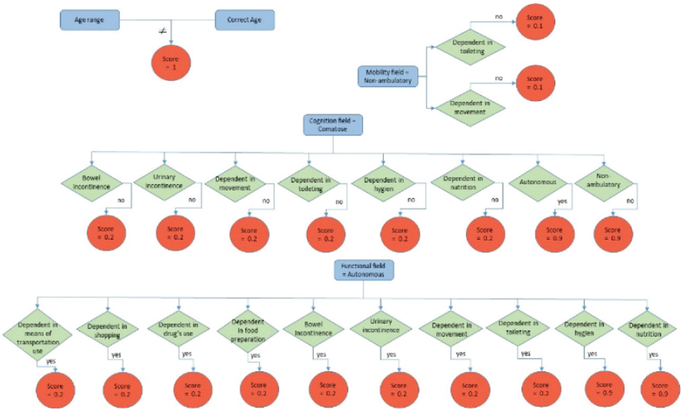


Fig. 1. Brass pipeline flowchart.

- If the *cognition field* is set to “comatose”, the *functional field* must contain all of the following choices: “dependent in nutrition”, “dependent in hygiene”, “dependent in toileting”, “dependent in movement”, “urinary incontinence” and “bowel incontinence”. For each missing choice, a score equal to 0.2 is added to the metric.
- Focusing on the *functional field*, if the choice “autonomous” is set, none of the choices described above should be reported. For this reason, if also “dependent in nutrition”, “dependent in hygiene”, “dependent in toileting”, “dependent in movement”, “urinary incontinence”, “bowel incontinence”, “dependent in food preparation”, “dependent in drug’s use”, “dependent in money’s use”, “dependent in shopping” and “dependent in means of transport’s use” are set in the field, for each choice a score equal to 0.2 is added to the metric.
- If the *mobility field* is set to “non-ambulatory”, in the functional field the choices “dependent in movement” and “dependent in toileting” should be present. For each missing choice, a score equal to 0.1 is added to the metric.

The maximum total score obtained with the above condition is equal to 6.4 and indicates a low agreement between the fields of the scale. All the aforementioned weights have been proposed by qualified medical personnel with extensive experience in managing hospitalized patients.

IDA Metric. As described above, the IDA scale is composed of 10 items: the final score is obtained by adding the value from each item and may range from 7 to 28. Similar to the Brass scale metric, clinical observations are used to define our pipeline to obtain the final score and evaluate the reliability of the scale’s measurement. The proposed pipeline is showed in Fig. 2:

- If the choice in the *origin field* is wrong, a score equal to 1 is assigned to the metric. This is the highest score since the value is easily obtainable from the hospitalization information.
- If the *sensory perception field* is set to “soporific state, coma”, the *mobilization field* will only be “non-ambulatory” or “armchair mobilization”. Different choices in *mobilization field* are evaluated with a score equal to 0.9.

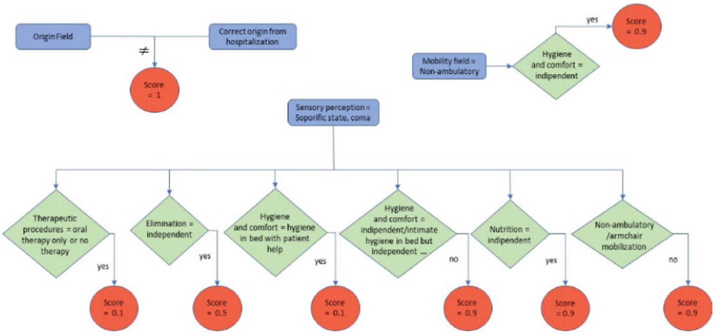

Origin Field
Correct origin from hospitalization
≠
Score = 1
Mobility field = Non-ambulatory
Hygiene and comfort = indipendent
yes
Score = 0.9
Sensory perception = Soporific state, coma
Therapeutic procedures = oral therapy only or no therapy
yes
Score = 0.1
Elimination = independent
yes
Score = 0.5
Hygiene and comfort = hygiene in bed with patient help
yes
Score = 0.1
Hygiene and comfort = indipendent/intimate hygiene in bed but independent ...
no
Score = 0.9
Nutrition = indipendent
yes
Score = 0.9
Non-ambulatory /armchair mobilization
no
Score = 0.9

Fig. 2. IDA pipeline flowchart.

- At the same time, if the *sensory perception field* is set to “soporific state, coma”, the *nutrition field* cannot be filled with the choice “independent”. If such situation occurs, a score equal to 0.9 is added to the metric.
- If the *sensory perception field* is set to “soporific state, coma”, the *hygiene and comfort field* cannot be “independent” or “intimate hygiene in bed but independent in the use of services”. For any of these choices, a score equal to 0.9 is added to the metric.
- Following the previous case, if the *sensory perception field* is set to “soporific state, coma”, the *hygiene and comfort field* cannot be “hygiene in bed with patient help”. In this case the score is lower and equals 0.1.
- If the *sensory perception field* is set to “soporific state, coma”, the *elimination field* cannot be “independent”. In this case the score is equal to 0.5.
- If the *sensory perception field* is set to “soporific state, coma”, the *therapeutic procedures field* cannot be “oral therapy only or no therapy”. In this case the score is equal to 0.1.
- Finally, if the *mobilization field* is set to “non-ambulatory”, in the *hygiene and comfort field* the choice “independent” is not correct. In fact, the other choices involve hygiene in bed, that is the proper condition if the patient is bedridden. For this case, a score equal to 0.9 is added to the metric.

The maximum total score obtained with the above condition is equal to 5.3 and indicates low agreement between the fields of the scale.

2.2 Overlap Analysis

Analyzing the two scales in detail, it is immediate to see that many fields provide the same information about the patient’s clinical condition. In fact, both scales analyze the dependency in some activity of daily living (ADLs), such as nutrition, hygiene, mobilization, elimination, and cognitive status. Exploiting this high overlap, it is possible to estimate the match among two scales compiled close in time.

For this reason, the pipeline in Fig. 3 is proposed. The first step is to individuate all the couples of IDA and Brass scales measured for the same patient within a maximum time interval of 2 h. This time range makes it possible to compare two scales that refer to the same clinical condition of the patient, avoiding different interpretations of cognitive or functional status. In addition, the analysis on the cohort of patients proposed in this study revealed that the two scales are often performed close together in time.

Six conditions are analyzed: nutrition, elimination, hygiene, mobilization, cognitive status and care intensity. Nutrition, elimination and hygiene conditions compare only the independent status, that in Brass scale is explained in *functional status*, whereas in IDA scale in *nutrition, elimination and hygiene* and *comfort fields*. Mobilization condition, instead, can be compared with three distinct cases; patient walks, not-walks or walks with help. These different choices are allowed in *mobilization field* in both scales and in this way it is possible to distinguish several clinical conditions. Also, there are more overlap cases between fields for the cognitive status, that in Brass is explained in *cognitive status field* and in IDA in *sensory perception field*. The overlap conditions are for comatose, oriented and disoriented. Disoriented condition is expressed in Brass scale with the choices "always disoriented, sedatives day and night" and "occasionally disoriented, with or without sedatives"; in IDA is expressed with the choices "occasional temporal-space disorientation (day and night sedatives)" and "constant disorientation temporal-space (sedatives day and night)". This consideration was made in order to avoid possible different interpretation and fast changes in patient clinical condition. Lastly, care intensity level can be easily compared between the scales. In fact, for each scale, three levels of intensity of care are possible, low, medium and high.

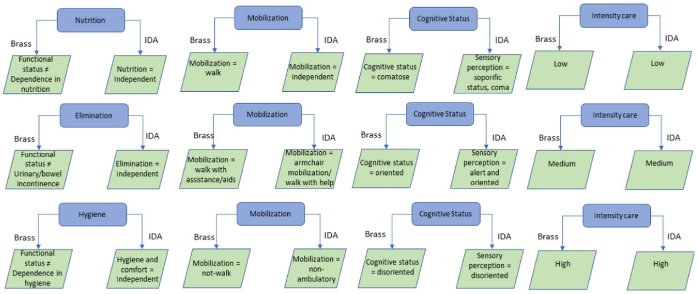

Nutrition
Brass
IDA
Functional status ≠ Dependence in nutrition
Nutrition = Independent
Mobilization
Brass
IDA
Mobilization = walk
Mobilization = independent
Cognitive Status
Brass
IDA
Cognitive status = comatose
Sensory perception = soporific status, coma
Intensity care
Brass
IDA
Low
Low
Elimination
Brass
IDA
Functional status ≠ Urinary/bowel incontinence
Elimination = Independent
Mobilization
Brass
IDA
Mobilization = walk with assistance/aids
Mobilization = armchair mobilization/ walk with help
Cognitive Status
Brass
IDA
Cognitive status = oriented
Sensory perception = alert and oriented
Intensity care
Brass
IDA
Medium
Medium
Hygiene
Brass
IDA
Functional status ≠ Dependence in hygiene
Hygiene and comfort = Independent
Mobilization
Brass
IDA
Mobilization = not-walk
Mobilization = non-ambulatory
Cognitive Status
Brass
IDA
Cognitive status = disoriented
Sensory perception = disoriented
Intensity care
Brass
IDA
High
High

Fig. 3. Overlap pipeline flowchart.

Table 1. Summary of the included variables in the predictive models for each scale.

Scale	Variables
IDA	Origin, Nutrition, Elimination, Hygiene and Comfort, Mobilization, Diagnostic Procedures, Therapeutic Procedures, Sensory Perception, intensity of care, value
Brass	Age, Life Condition, Number of Previous Hospital Admissions, Number of actual clinical problems, Cognitive Status, Functional Status, Behavioral Model, Mobilization, Sensory Deficits, Number of Drugs Taken, intensity of care, value

For each explained condition, the possible overlap is evaluated. Lower percentage of overlapping represent lower concordance between the clinical condition reported in the two scales.

3 Results

In the following Sections, we report the obtained results about the prediction model.

3.1 Prediction Model

The aim of this study is to predict the value and the intensity of care of one scale starting from the information of the other one. Four predictive models are analyzed to consider all possible combination, i.e. Brass scale that predicts value of IDA scale, Brass scale that predicts intensity of care of IDA scale, IDA scale that predicts value of Brass scale and IDA scale that predicts intensity of care of Brass scale. This analysis was performed in Orange.

70% (13.041 measurements couples) and 30% (5.589 measurements couples) of the samples are split into training and test cohorts. Logistic regression (LR), decision tree (DT) and random forest (RF) algorithm with five-fold cross-validation (CV) technique are implemented to estimate the intensity of care, whereas only the last two for the scale value. Variables used in the algorithms are listed in Table 1. All variables are categorical expect for the value's scale.

IDA Scale Predicts Brass Scale. Two models are developed to predict intensity of care and value of the Brass scales. The developed models are tested with the test set. As shown in Fig. 4 (a)–(c), for each intensity of care class the powerful predictive capacity is evidence by AUC in three basic models. For simplicity, average of high, medium and low scale AUC obtained are reported: LR = 0.984; DT = 0.937; RF = 0.978. In this case, the high and low intensity of care, Fig. 4 (a) and (c) respectively, are easier

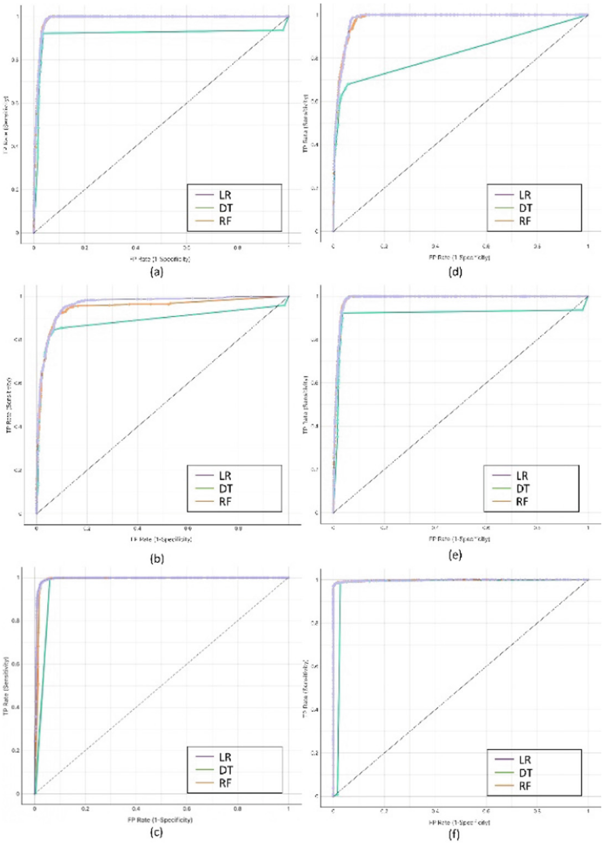


Fig. 4. ROC curves obtained from the prediction models. IDA scale predicts Brass scale: (a) High, (b) Medium and (c) Low intensity of care. Brass scale predicts IDA scale: (d) High, (e) Medium and (f) Low intensity of care.

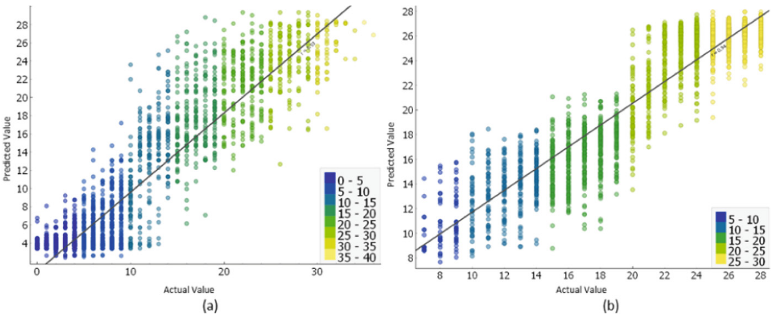


Fig. 5. Scatter plot of actual values and predicted ones obtained with RD model: (a) Brass scale values and (b) IDA scale values.

to predict compare the medium level (Fig. 4 (b)). DT and RF models are also used to predict Brass value and to evaluate the algorithms, root mean square error is calculated. RMSE obtained with DT is equal to 2.665 that is higher than it obtained with RF, that is 2.606. Figure 5 (a) shows the scatter plot of actual values and predicted ones obtained with RF model.

Brass Scale Predicts IDA Scale. Also in this case, two models are developed to predict intensity of care and value of the IDA scales. The developed models are tested with the test set. In Fig. 4 (d)–(f) the AUC for each intensity care is showed, with the comparison of the three predictive models. For simplicity, average of high, medium and low scale AUC obtained are reported: average AUC LR = 0.992; DT = 0.934; RF = 0.992. In this case, the medium and low intensity of care, Fig. 4 (e) and (f) respectively, are easier to predict compare the high level (Fig. 5 (d)). Specifically, the DT model result is very lower than the other ones (AUC LR = 0.980; DT = 0.822; RF = 0.980). Predicted IDA values by the DT and RF models are evaluated with RMSE. Following the previous results, RMSE obtained with DT higher than it obtained with RF, and are equal to 1.855 and 1.669, respectively. Figure 4 (b) shows the scatter plot of actual values and predicted ones obtained with RF model.

4 Conclusions

Nursing scales represent one of the evaluation tools that allows to quantify patient's clinical condition in an immediate way. In this study we analyze two different scales, Brass and IDA, with the twofold purpose of develop a metric to assess the quality of a single measurement scale and of create a predictive model to estimate one scale using the information of the other one. Preliminary results shown the possibility to create a generic metric that evaluates the concordance of the information selected in each scale. In this way, is possible to create a database of reliable measurements for the proposed predictive model. That model shows promising results to evaluate one scale from the other one. The goal of this work is not only to demonstrate that one scale can be predicted from the other one, but also to provide a possible tool that allows clinicians to save time in filling out the scales and focus more on clinical work.

References

1. Blaylock, A., Cason, C.L.: Discharge planning: predicting patients' needs (1992)
2. Cavaliere, B.: Misurare la complessità assistenziale: Metodi e strumenti per le professioni sanitarie. Maggioli, Rimini, Italy (2009)
3. Giovannini, S., et al.: A new model of multidimensional discharge planning: continuity of care for frail and complex inpatients. Eur. Rev. Med. Pharmacol. Sci. **24**, 13009–13014 (2020)
4. Johnson, S.G., Speedie, S., Simon, G., Kumar, V., Westra, B.L.: A data quality ontology for the secondary use of EHR data. In: AMIA Annual Symposium Proceedings, vol. 2015, p. 1937. American Medical Informatics Association (2015)
5. Ozonze, O., Scott, P.J., Hopgood, A.A.: Automating electronic health record data quality assessment. J. Med. Syst. **47**(1), 23 (2023)
6. R Core Team, R., et al.: R: A language and environment for statistical computing (2013)
7. Savino, M., et al.*:* An interactive dashboard for patient monitoring and management: a support tool to the continuity of care centre. In: Juarez, J.M., Marcos, M., Stiglic, G., Tucker, A. (eds.) AIME 2023. LNCS, vol. 13897, pp. 368–372. Springer, Cham (2023). https://doi.org/10.1007/978-3-031-34344-5_45
8. Wang, B., Lai, J., Jin, F., Liao, X., Zhu, H., Yao, C., et al.: Clinical source data production and quality control in real-world studies: proposal for development of the esource record system. JMIR Res. Protoc. **11**(12), e42754 (2022)
9. Weiskopf, N.G., Bakken, S., Hripcsak, G., Weng, C.: A data quality assessment guideline for electronic health record data reuse. Egems **5**(1) (2017)

Process Mining for Capacity Planning and Reconfiguration of a Logistics System to Enhance the Intra-Hospital Patient Transport. Case Study

Tobias Kropp[1](✉), Shiva Faeghi[2], and Kunibert Lennerts[1]

[1] Institute for Technology and Management in Construction, Karlsruhe Institute of Technology,
Gotthard-Franz-Str. 3, 76131 Karlsruhe, Germany
tobias.kropp@kit.edu

[2] Health Technology and Services Research, University of Twente, Drienerlolaan 5,
7500 AE Enschede, The Netherlands
https://www.tmb.kit.edu/english/index.php

Abstract. Intra-hospital patient transport (IHPT) service is one of the important contributors to efficiency in hospitals due to its high prevalence. The efficiency of this service is, in turn, dependent on proper planning of capacities and resources. Although there is extensive research focusing on improving capacity planning, there is little research available on posterior analysis of real-life executions of transport activities and evaluation methods. Therefore, this paper first provides a set of Key Performance Indicators to measure the efficiency of IHPT services using process mining approaches. Second, it conducts an extensive multidimensional analysis to support capacity planning by examining data containing various event- and case-specific information from IHPT process for a period of 42 months beginning from January 2019 in a German hospital. Different perspectives are considered to enable multidimensional analysis and provide insights regarding the behavior of different elements involved in the transport process. Daily and hourly assignments are evaluated to investigate transport capacities, activity intervals, automatically and manually dispatched assignments, as well as the most significant routes concerning transport delays. The analysis showed that 34.2% of the transports experienced delays of ten or more minutes. After identifying the causes of these delays and process bottlenecks, several technical and operational solutions are proposed, which are evaluated by domain experts in the case hospital. This paper shows the capability of process mining methods to provide holistic and clear insights into processes, which can help hospitals better understand the organization of their processes and address the challenges outlined in IHPT services.

Keywords: Process Mining · Logistical Processes · Patient Transport · Capacity Planning · Process Efficiency

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 138–150, 2024.
https://doi.org/10.1007/978-3-031-66538-7_15

1 Introduction

Capacity planning is an essential process in hospitals to ensure that resources are managed efficiently and that quality of care meets high standards. Planning of resources for logistic activities includes not only material flows, but also transport of patients within the hospital. Intra-hospital patient transport (IHPT) is a challenging task, since it involves medical aspects and requires coordination between different functional areas to prevent medical complications for patients and avoid long waiting times [4]. Due to its broad use, IHPT plays a crucial role in providing efficient and timely medical treatments [5, 7, 16]. IHPT refers to the internal patient transfer within a hospital, e.g. between different functional areas and wards [12]. The effectiveness of this service and its associated processes have major impacts on both patient satisfaction and clinical outcomes [5].

To support the planning of IHPT capacity in a hospital, it is necessary to analyze to what extent the capacity used in logistic activities meets transport requirements. Process mining can help identify bottlenecks and inefficiencies in processes [17], which can help hospitals accordingly improve their capacity allocation.

Generally, [11] worked out ten specific challenges to using process mining in the healthcare domain, of which our work addresses the following: discover beyond discovery, deal with reality, pay attention to data quality, and take care of privacy and security.

With the help of Process Mining, we examine the IHPT process in a German hospital, covering a period of approximately 3.5 years, from different perspectives, and conduct multidimensional analysis. The goal is to examine which decisions or parameters were favorable or disadvantageous based on particular key performance indicators (KPIs). The remainder of the paper is organized as follows: Sect. 2 reviews literature on the application of different approaches to design and organize IHPT processes in hospitals. Section 3 gives an overview of our research questions and objectives. The IHPT investigation is conducted in Sect. 4. This section discovers the process models, process bottlenecks, examines the related KPIs and gives tailored improvement proposals. Finally, the paper concludes with limitations and future work in Sect. 5.

2 Related Work

Different quantitative approaches such as mathematical programming or simulation models have been developed to design and evaluate IHPT. For example, [14] used statistical methods and a mixed-integer model to determine the best distribution of employees on the most popular routes to reduce the completion time of patient transport activities. [6] proposed a computer-based planning system based on fast heuristics to provide an efficient and timely IHPT service. [10] developed an integer programming model and a discrete event simulation to standardize IHPT processes and improve the planning of the transport staff.

Mathematical programming models are mostly based on closed-form expressions developed for case-specific problems [15]. Therefore, they are sensitive to the level of detail and complexity of the problem under study [15]. They often require many assumptions and therefore hardly represent real-life processes [19]. In contrast, simulation models are less sensitive to complexity [15] but the problem is the high modeling effort, which is one of the barriers to the application of simulation by healthcare managers [9].

In recent years, process mining has been introduced to overcome these limitations, although it tends to focus on historical process data. However, it can also be used as input for prescriptive approaches. Process mining can help to discover, analyze, and improve processes from a large amount of data stored in an information system. [13] analyzed 263 articles on process mining in healthcare and emphasizes that only about 7.6% of the articles deal with analysis of organizational processes and only another 15.2% deal with organizational processes in part. [2, 18, 20] developed DES models on the basis of process mining analyzes to optimize organizational health care processes and, thus, also capacity management. However, there is little or no reported research on the application of process mining to plan resources in IHPT services. In [8], a process discovery and conformance analysis was performed on an IHPT data set of one year to show the capability of process mining analysis on event data captured in the hospital logistics system. However, the solutions presented are in an early stage and capacity analyses were not applied or tested.

3 Research Questions and Objectives

There are three mainly established subareas in process mining, namely process discovery, process conformance, and process enhancement [3, 17]. This paper will focus on providing specific solutions for the enhancement of the IHPT process and capacity planning and will underpin them with observations from the process discovery and conformance analysis of a data set of 3.5 years. Our research questions are defined as follows:

RQ 1: How can mining in real-life data sets with multiple case- and event-related attributes help to analyze capacity planning in IHPT beyond process discovery?
RQ 2: What are the important factors and limitations to consider when proposing capacity improvement measures in IHPT based on real-life data sets with multiple attributes related to case and event?

This paper illustrates the substantial potential of process mining techniques to analyze the IHPT process and to propose concrete process improvement ideas (i.e., process enhancement) based on previously derived KPIs taking into account different perspectives, time periods, data quality aspects, and data privacy requirements.

4 Patient Transport Analysis

The insights and ideas presented in this section were discussed in multiple sessions with hospital process managers and thus validated to reflect reality. Their assessment is incorporated into the analysis. For the analysis, the *Celonis® Execution Management System*, which can generate simple visualizations using Directly-Follows Graphs (DFGs), is used. DFGs typically do not consider concurrencies and causalities [1]. To ensure proper sequencing in the case of equal event timestamps, a unique sorting number for each event is added based on the activity executed during data preparation in consultation with the process managers. COVID-19 also had an influence on the process, but the investigations in this paper will be on an aggregated level over a longer timespan and therefore not COVID-19-specific.

4.1 General Process Information from Different Perspectives

The data analyzed covered transports where the first activity occurred within the period from 01/01/2019 to 30/06/2022, which reflects a period of around 182.5 weeks (a handful of transport cases had some activities on July 1, 2022, although their first registered activity took place beforehand). Only transports that were completed fully and in which patients were transport subjects were examined, and there was no additional special service in addition to transport. To conduct the analysis, first the assignment ID of a transport (i.e. unique number for each individual transport), and second, the transporter ID related to the transport is considered as the case ID to gain insights into the transport processes. A transporter ID is understood to be a combination of a specific number of a unique mobile device (that receives transport assignments) and the date on which the specific device appeared in the logs so that a unique transporter ID can be derived and equated with a transporter on a specific day.

Other information, like the patient ID, can allow one to examine the processes from the perspective of the patients. However, this will not be the scope of this paper.

Assignment ID is Case ID. There were 256,266 patient transport cases conducted and completed, resulting in an average of 3.7 transports per patient (there were 69,810 unique patient IDs in the data set). After being requested, transports can be assigned manually (by an employee) or automatically (directly by the logistics system). After selecting a transporter, the transport assignment is forwarded to the transporter device. However, 34.2% of all transports experienced delays of ten or more minutes, that is, the transport is completed ten or more minutes after the pre-planned completion time. This limit, above which transports are classified as significantly delayed, was defined by process managers. The event log examined contains different activities. Table 1 in the appendix shows in how many cases logged activities occur and also how often the activities occur in total over all cases (activities can occur multiple times within a case).

Transporter ID is Case ID. On the whole, there were 10,089 different transporter IDs (combination of a unique device number being present on an individual day) involved in the patient transport within the hospital, each representing a transporter on a specific day. On average, approximately ten different transporters from Monday to Friday and about three different transporters from Saturday to Sunday participated in the transports per day and there were 25.4 transports per day per transporter.

4.2 Process Analysis - Deep Dive

In this subsection, a deeper analysis is made from the point of view of the transports (a transport's assignment ID is the case ID) and multiple KPIs are presented. First, the different variants of the process are considered. Then, activity intervals are examined and critical transport routes are highlighted. Subsequently, analyses are carried out with regard to the assignment situation during the course of the day and the weekday to allow comparisons with the transport capacities provided by the hospital to derive recommendations for action.

Process Variants and Activity Intervals. Within the 256,266 patient transport cases, there are 1,977 different variants in the activity control flow. The main variant (left) and the first 9 variants (right) together are shown in Fig. 2 in the appendix. Figure 2 also shows the delay rates of the respective process variants. *Variant 1* reflects the main variant that covers about 45% of all cases and consists of eight activities. The activity sequence is: "transport request", "waiting list for commissioning", "assignment sent to device", "assignment accepted", "arrival at pick-up location", "transport started", "arrival at target location", "transport completed". The first nine variants together show loops and variations exclusively in the first half of the process. *Variant 4* (see left model in Fig. 3) and *Variant 8* (see right model in Fig. 3 in the appendix) stand out in particular because of their increased rate of delays. Since detailed event-specific information is also included in addition to the case-specific attributes that provide information about the case as a whole, a root cause analysis can be performed within the two variants on an event level in addition to the analysis of general activities and throughput times. It is noticeable that in *Variant 4* it is mainly the failure of the transporter to react that leads to a new request for transport before the same process flow like that of the first variant starts. *Variant 4* lead to a delay probability greater than 61% in the subsequent processing of the transport assignment. *Variant 8* shows the same behavior as *Variant 4*, but in *Variant 8* there is also the step "transport is preregistered" as the first activity. A transport can be pre-registered if, e.g. it is a return transport that is required after an initial transport has been requested. Adjustment ideas to reduce the delay rate are given after examining the relevant activity intervals.

In general, pre-registering assignments decreases the delay rate. This can be read from the delay rates in Fig. 2 in the appendix. *Variant 2* has the activity "transport is preregistered" before the activity "transport request" and flows further like *Variant 1. Variant 2* has a slightly lower delay rate than *Variant 1. Variant 1* and *Variant 2* reflect the planned process flow, while all other variants are considered non-conforming in discussion with process managers.

Between the activity "assignment sent to device" and the acceptance of the assignment by the transporter ("transport accepted"), on average it takes about two to three minutes and even zero minutes in the median, which means that the acceptance of the assignment happens in most of the cases instantly (89% of all cases are accepted within zero and three minutes after being sent to a device). This observation was also confirmed by the process managers, who have experience in the field that transporters are already accepting new assignments while executing previous ones. If a transporter does not respond and the assignment has to be re-requested, it almost always leads to delays (see the delay rates in Fig. 3 in the appendix). To reduce this problem, the buffer time between "transport received at end device" and a transporter not responding, which leads to a re-assignment, is to be reduced. In the future, the appropriate time interval appears to be around three minutes, since 89% of all cases are currently accepted within zero and three minutes after being sent to a device. In approximately 80% of all transports (transports of all variants), the interval between the "transport request" and the "waiting list for commissioning" takes between zero and eight minutes. If it takes between zero and four minutes, the resulting delay rate for the transports is around 25%. If it takes between four and eight minutes, the resulting delay rate is already around 40%. If more than eight minutes elapse between the "transport request" and the "waiting list for commissioning", the resulting delay rate for transports is 66%. An alarm could be established if more than, for example, four minutes elapse between "transport request" and "waiting list for commissioning" to promptly re-request the transports and keep delay rates as low as possible.

Data show a delay already at the patient's pick-up location in about 50% of all cases. Compared to the planned end time of transports, around 34% are still delayed by the time a transport is logged as completed (see Subsect. 4.1). This shows that overall, despite delays at the pick-up location, certain transports manage to arrive on time at the arrival point and that the later activities, which in turn represent the accompanied transport, tend to be classified as less critical in terms of throughput times or delays. This raises questions about the efficient planning of transport capacities and distribution of transports, including the initial activities up until "arrival at pick-up location". Here, it should be mentioned that data quality and thus delay rates depend on the manual confirmation of individual work steps by the transporters via their devices. Therefore, they may not always reflect the correct timestamp at which an activity was performed. To make the data more reliable in the future, it might also be useful to introduce quick response (QR) code or radio frequency identification (RFID) scanning at all relevant locations (pick-up and target locations) in the hospital to precisely capture when the transporters fulfill their process steps.

Critical Routes. Table 2 in the appendix presents the most critical transport routes in terms of their delay time at the end of the transport process. They are sorted by the total number of delayed minutes that occurred on each route. The total delays are calculated by multiplying the average delay per case on each transport route by the number of delayed cases. Only transports that had a delay of at least ten minutes are considered for this calculation. The ranking according to the highest sum of minutes of delay on the respective routes, which results from the combination of the average delay time per transport and the number of delayed transports, ensures that Table 2 in the appendix reflects the most problematic routes, in general, in all the cases considered. In particular, the route between a ward "Station A4.2" and a functional area "Endoscopy" (both in the same building House A) stands out in the first (one transport direction) and fourth (reverse transport direction) rows. In general, it is recommended to increase the scheduled throughput times for transports on this route and also on the other routes identifiable in Table 2 in the appendix.

Capacity Evaluation. Delayed assignments are more common on Mondays and Fridays than on the other days of the week. Generally on Mondays, the most transport activities take place. On Saturdays and Sundays, there are fewer transports per day than in the rest of the week, and the delay rate is lower during the weekend. Table 3 in the appendix shows the statistics for all weekdays.

Delays occur most frequently between 10:00 and 12:59. A cause for these delays can be found in the event-specific data and is because the transporters' devices are full and assignments cannot be sent. This observation starts already increasing from 07:00. There are also many delays between 09:00 to 09:59 and 13:00 to 13:59. Mainly, between 09:00 and 13:59, either the number of requestable planned starts should be regulated, or more transport capacities (i.e., transporters) should be available. In addition, there are peaks in relatively delayed cases between 06:00 and 07:00. Since the number of cases is not very high during this time, it seems more meaningful to increase the transport capacities rather than to limit the number of assignments. Table 3 in the appendix shows that even though on Mondays there is the highest number of transports per day, the number of transporters is just the fourth highest compared to the other days of the week. Figure 1 shows a comparison of planned transports by hour and the corresponding transporters available who started transports at specific times, using Monday as an example. The same comparisons could be made equally for all other days of the week.

Figure 1 shows that there is a peak in assignments between 10:00 and 10:59. At the same time, there is a peak of approximately 4.6 assignments per transporter. To avoid a backlog of assignments, transportation capacity should be increased in the morning (until 12:00/13:00), as the assignment peak is reached around 11:00 and then the absolute number of assignments per hour decreases. For a lower delay rate, there should generally be no more than three to four assignments per available transporter per hour (a transporter completes his transport task in 19–20 min on average or 14 min on median, from acceptance to the end of the transport) to reduce delays caused by backed up assignments

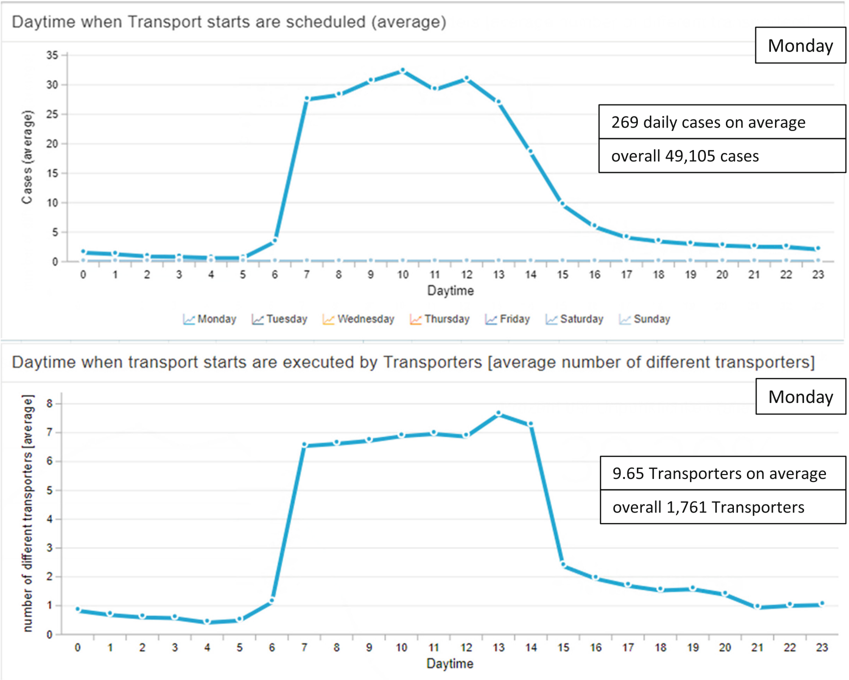


Fig. 1. Comparison of planned transports (upper part) and available transporters (lower part) on average on Mondays (the minutes of each timestamp are rounded down to zero, e.g., 14:49 becomes 14:00) (adopted from *Celonis*®).

in the afternoon. These observations were made equally on the days from Tuesday to Friday, and the solutions mentioned should also be applied on these days. In Table 3 in the appendix it can be observed that fewer transporters are active on Mondays and Fridays than on other days of the week, despite the highest delays. However, due to the highest number of assignments on Mondays, a slight increase in transportation capacity should be considered. The load of the transporters is lower on the weekend, with a peak of 3.9 transports per unit per hour on Saturdays and 2.8 on Sundays than during the week (see Table 3 in the appendix). This KPI certainly seems to have a major influence on the delay rate. Furthermore, the delay rate of the automatically assigned transports (by the automatic dispatching system), with 30%, is slightly better than the delay rate of the manually assigned transports (by an employee), with 36%. But the automatic dispatching system only handles less than half of the cases compared to manual dispatchers and works mainly at times when there are fewer transports requested (usually outside of core working hours and on weekends). The way in which manually and automatically assigned transports were put together seemed to influence delayed cases, too. The more both were conducted in parallel, the more delayed cases resulting from automatically assigned transports could be observed. From this it can be inferred that the automatic system and the manual dispatcher should operate only in a timely separated manner.

5 Conclusion and Future Work

The investigations show the distribution of transportation assignments and the availability of capacity at times of the day and days of the week. Therefore, capacity evaluations are performed and improvement ideas are developed through root cause analysis. The average peak number of requested transports per available transporter per hour needs to be reduced for less delayed transport cases.

In addition, the automatic dispatching system and the manual dispatcher should operate at completely separate times (not in parallel) so that both can efficiently contribute to less delayed transport cases. However, an investigation of how many orders are open at any given time could be interesting for further interpretation. For this purpose, in the future it should be examined on every day of the week and every hour how many orders have been requested and which have not yet been closed. These necessities can also be transferred to other hospitals. Discussions with domain experts ensure the validity of the investigations and help them understand and interpret process data effectively.

Data such as the number of transport orders that are on the transporters' devices at the respective times when individual transports are executed (currently up to three assignments per device are possible) are not reflected in the data. Therefore, adjustment ideas towards the maximum number of receivable transports per transporter cannot be supported with historical information at this point. Data management should be improved accordingly.

It is generally noticeable that the possible adjustments that can be made seem relentless and that it is difficult to commit to specific measures. Many of the possible solutions result of individual attribute filtering of the analysis and custom KPI evaluations. Of course, these depend highly on the know-how of the data analysts and the input of the domain experts. It is now necessary to transfer individual solution approaches into practice and to check the results with analogous analysis, as has been done in this paper, on the basis of the identified KPIs. However, it would be very helpful to use e.g. simulations to predict the resulting KPI developments after adjusting the parameters of processes. It is the goal to develop prediction models on the basis of historical data and expert knowledge. Such models could support decision making in process change management by running several adaptation variants before practical implementations, and the best variant could be transferred to practice with data support.

Acknowledgments. Supported and funded by the case study hospital and by the German Research Foundation (DFG) - project number 456684366.

Data Availability Statement. Data sharing is not applicable to this article due to data protection regulations.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

Appendix

Table 1. Statistics on all activities. They are sorted by case count and within the same case count, furthermore alphabetically according to the German activity name.

Activity (German - in system)	Activity (English translation)	case count	activity count
Auftrag abgeschlossen	transport completed	256,266	257,205
Auftrag an Endgerät	assignment sent to device	256,266	282,505
Auftrag angenommen	assignment accepted	256,266	262,262
Transport begonnen	transport started	256,266	256,556
Warteliste Kommissionierung	waiting list for commissioning	256,266	302,271
an Abholort	arrival at pick-up location	256,265	257,334
an Ankunftsort	arrival at target location	256,266	256,376
Anforderung	transport request	256,261	317.641
Vorgemerkt	transport is pre-registered	111,906	127,579
Verfall - nicht zugestellt	expiration - not delivered	6,021	9,714
Dispo	transport assignment	145	153
Storniert	transport canceled	33	39

Table 2. Critical routes (only delayed cases) that lead overall to a high sum of delayed minutes.

pick-up house	pick-up location	target house	target location	delayed cases	avg. delay per case [min]	sum of delay [min]
House A	Station A4.2	House A	Endoscopy	2,871	37.68	108,175.80
House B	Station B2.2	House H	Radiology	1,776	40.55	72,012.82
House A	Station A2.1	House A	FUDI EKG	1,972	36.09	71,538.31
House A	Endoscopy	House A	Station A4.2	1,492	39.99	59,664.18
House B	Emergency Dpmt	House A	Station A4.2	1,267	37.25	47,191.63
…	…	…	…	…	…	…

Table 3. Weekday statistics.

weekday	absolute cases per day	average cases per day	average number of transporters per day	average peak number of transports requested per available transporter per hour	rate of delayed cases [%]
Monday	49,105	269	9.65	4.6	39.30
Tuesday	47,484	260	10.31	4.4	32.55
Wednesday	48,364	265	10.55	4.2	34.98
Thursday	45,286	248	10.18	4.3	31.41
Friday	44,532	244	9.44	4.6	37.70
Saturday	11,641	64	3.25	3.9	23.74
Sunday	9,854	54	3.16	2.8	22.61

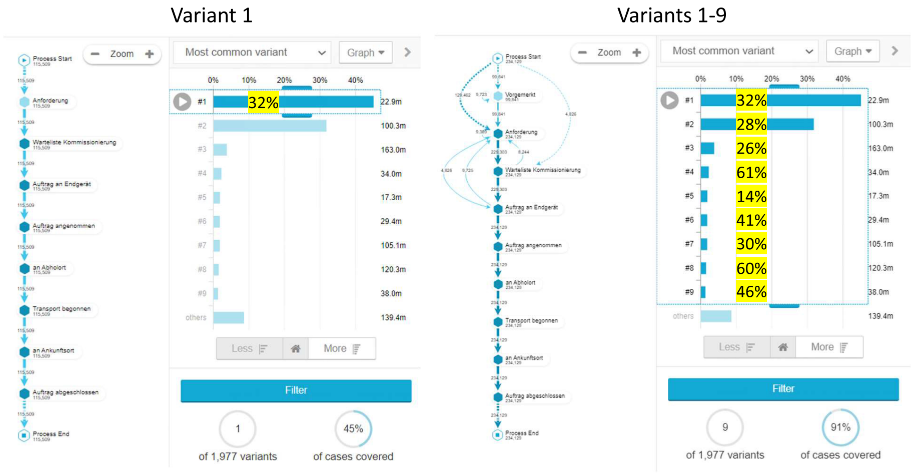


Fig. 2. Process model of the most frequent variant (representing 45% of all cases) on the left and process model of the first 9 variants (representing 91% of all cases) on the right with case counts on paths and activities. Delay rates of the respective process variants are represented in yellow highlighted percentages. Bar charts reflect relative frequency and the numbers to the right of the bar charts reflect the median throughput time of the variants (adopted from *Celonis*®). (Color figure online)

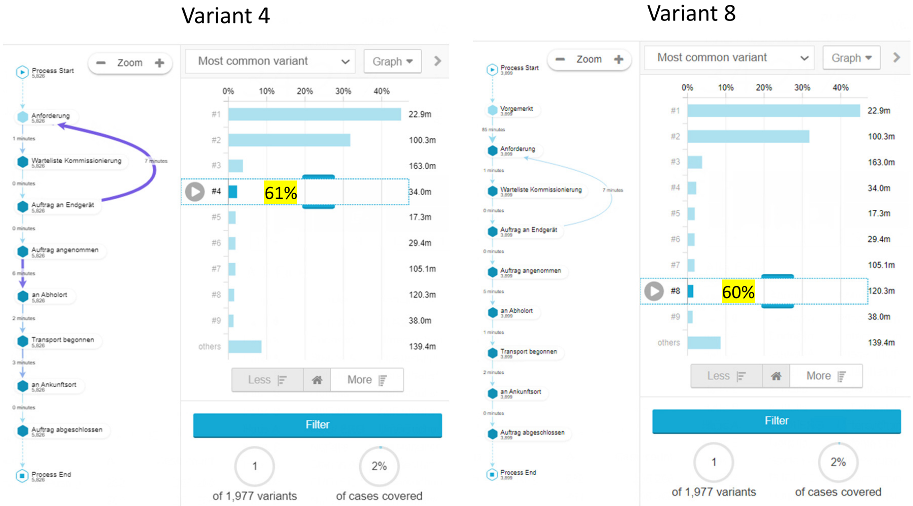


Fig. 3. Process model of the variant 4 (left model) and variant 8 (right model) with median throughput times on the paths and case counts on the activities. Delay rates of both variants are represented in yellow percentages. Bar charts reflect relative frequency and the numbers to the right of the bar charts reflect the median throughput time of the variants (adopted from *Celonis*®). (Color figure online)

References

1. van der Aalst, W.M.P.: Academic view: development of the process mining discipline. In: Reinkemeyer, L. (eds.) Process Mining in Action, pp. 181–196. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-40172-6_21
2. Abohamad, W., Ramy, A., Arisha, A.: A hybrid process-mining approach for simulation modeling. In: Chan, W.K., D'Ambrogio, A., Zacharewicz, G., Mustafee, N., Wainer, G., Page, E.H. (eds.) WSC 2017, Piscataway, NJ, pp. 1527–1538. IEEE (2017). https://doi.org/10.1109/WSC.2017.8247894
3. van der Aalst, W.M.P., et al.: Process mining manifesto. In: Daniel, F., Barkaoui, K., Dustdar, S. (eds.) BPM 2011. LNBIP, vol. 99, pp. 169–194. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-28108-2_19
4. Beaudry, A., Laporte, G., Melo, T., Nickel, S.: Dynamic transportation of patients in hospitals. OR Spectr. Quant. Approaches Manag. **32**(1), 77–107 (2010). https://doi.org/10.1007/s00291-008-0135-6
5. Beckmann, U., Gillies, D.M., Berenholtz, S.M., Wu, A.W., Pronovost, P.: Incidents relating to the intra-hospital transfer of critically ill patients. Intensive Care Med. **30**(8), 1579–1585 (2004)
6. Hanne, T., Melo, T., Nickel, S.: Bringing robustness to patient flow management through optimized patient transports in hospitals. Interfaces **39**(3), 241–255 (2009)
7. Hendrich, A.L., Nelson, L.: Intra-unit patient transports: time, motion, and cost impact on hospital efficiency. Nurs. Econ. **23**(4), 157 (2005)
8. Kropp, T., Faeghi, S., Lennerts, K.: Evaluation of patient transport service in hospitals using process mining methods: patients' perspective. Int. J. Health Plann. Manag. **38**(2), 430–456 (2023). https://doi.org/10.1002/hpm.3593
9. Lowery, J.C.: Introduction to simulation in health care. In: Proceedings of the 28th Conference on Winter Simulation, pp. 78–84 (1996)
10. Mikaeili, M., Lam, S.S., Yao, J., Bosire, J.: Hospital patient transport workflow and staffing optimization. In: IIE Annual Conference. Proceedings, pp. 1546–1551. Institute of Industrial and Systems Engineers (IISE) (2019)
11. Munoz-Gama, J., et al.: Process mining for healthcare: characteristics and challenges. J. Biomed. Inform. **127**, 103994 (2022). https://doi.org/10.1016/j.jbi.2022.103994
12. Nakayama, D.K., Lester, S.S., Rich, D.R., Weidner, B.C., Glenn, J.B., Shaker, I.J.: Quality improvement and patient care checklists in intrahospital transfers involving pediatric surgery patients. J. Pediatr. Surg. **47**(1), 112–118 (2012). https://doi.org/10.1016/j.jpedsurg.2011.10.030
13. de Roock, E., Martin, N.: Process mining in healthcare - an updated perspective on the state of the art. J. Biomed. Inform. **127**, 103995 (2022). https://doi.org/10.1016/j.jbi.2022.103995
14. Séguin, S., Villeneuve, Y., Blouin-Delisle, C.H.: Improving patient transportation in hospitals using a mixed-integer programming model. Oper. Res. Health Care **23**, 100202 (2019)
15. Sinreich, D., Marmor, Y.: Emergency department operations: the basis for developing a simulation tool. IIE Trans. **37**(3), 233–245 (2005)
16. Ulrich, R.S., Zhu, X.: Medical complications of intra-hospital patient transports: Implications for architectural design and research. HERD **1**(1), 31–34 (2007). https://doi.org/10.1177/193758670700100113
17. van der Aalst, W.M.P.: Process Mining: Data Science in Action, vol. 2. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-49851-4
18. van Hulzen, G., Martin, N., Depaire, B., Souverijns, G.: Supporting capacity management decisions in healthcare using data-driven process simulation. J. Biomed. Inform. **129**, 104060 (2022). https://doi.org/10.1016/j.jbi.2022.104060

19. Zeltyn, S., et al.: Simulation-based models of emergency departments: operational, tactical, and strategic staffing. ACM TOMACS **21**(4), 1–25 (2011)
20. Zhou, Z., Wang, Y., Li, L.: Process mining based modeling and analysis of workflows in clinical care - a case study in a Chicago outpatient clinic. In: 2014 IEEE 11th International Conference on Networking, Sensing and Control (ICNSC 2014), Piscataway, NJ, pp. 590–595. IEEE (2014). https://doi.org/10.1109/ICNSC.2014.6819692

Radiotherapy Dose Optimization via Clinical Knowledge Based Reinforcement Learning

Paul Dubois[1,2](✉), Paul-Henry Cournède[1], Nikos Paragios[2], and Pascal Fenoglietto[3]

[1] Biomathematics, MICS, CentraleSupélec, Université Paris-Saclay, Paris, France
{p.dubois,paul-henry.cournede}@centralesupelec.fr
[2] TheraPanacea, Paris, France
{p.dubois,n.paragios}@therapanacea.eu
[3] Institut du Cancer de Montpellier (Val d'Aurelle), Montpellier, France
{paul.dubois,pascal.fenoglietto}@icm.unicancer.fr

Abstract. A radiation therapy plan finds an equilibrium between goals with no universal prioritization. The delicate balance between multiple objectives is typically done manually. The optimization process is further hindered by complex mathematical aspects, involving non-convex multi-objective inverse problems with a vast solution space. Expert bias introduces variability in clinical practice, as the preferences of radiation oncologists and medical physicists shape treatment planning. To surmount these challenges, we propose a first step towards a fully automated approach, using an innovative deep-learning framework. Using a clinically meaningful distance between doses, we trained a reinforcement learning agent to mimic a set of plans. This method allows automatic navigation toward acceptable solutions via the exploitation of optimal dose distributions found by human planners on previously treated patients. As this is ongoing research, we generated synthetic phantom patients and associated realistic clinical doses. Our deep learning agent successfully learned correct actions leading to treatment plans similar to past cases ones. The incapacity to reproduce human-like dose plans hinders adopting a fully automated treatment planning system; this method could start paving the way towards human-less treatment planning system technologies. In future work, we hope to be able to apply this technique to real cases.

Keywords: Radiotherapy · Dose Optimization · Reinforcement Learning · Deep Learning

1 Introduction

In contemporary radiation therapy, photon intensity modulated radiation therapy (IMRT) is a pivotal technique to attain precise and conformal dose distributions within target volumes [18]. This achievement owes its realization to the advent of the multileaf collimator (MLC) [5]. Radiation therapy is now a reliable treatment for oncology [14]. Despite this consensus, the way to deliver radiotherapy for its best result remains very dependent upon doctors. Moreover, there appears to be a large variability across physicians and centers, in terms of 3D structures contouring and irradiation, constrains priorities [3].

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 151–160, 2024.
https://doi.org/10.1007/978-3-031-66538-7_16

To achieve the best treatment, doctors must solve a complex inverse mathematical optimization problem with multiple trade-offs [10, 15]. However, a lack of standardized prioritization of constraints makes the optimization a real challenge. The standard procedure nowadays is to guide computer optimization manually: dosimetrists manually update the settings of an optimizing software so-called Treatment Planning System (TPS) [1].

There have been many tries to create a metric that quantifies the quality of a treatment plan, such as Normal Tissue Complication Probabilities (NTCP), target coverage, conformity index, and heterogeneity index, among others/to name a few [7, 8]. However, they have yet to satisfy all radio-oncologists, and the only reliable way to assess a doctor's plan is to evaluate the dose-volume histograms (DVHs) themselves.

As a result, Pareto surface exploration is unsuitable due to the lack of impartial quantitative measurement for a particular plan [6]. Other meta-optimization techniques are similarly bounded for the same reason [16, 17]. An extra challenge to attend for those is the fact that not all cases have the same difficulty. Hence, for an "easy" case, doctors will require an excellent dose (in terms of the metrics mentioned above), while they can be more permissive for "harder" cases. The context-aware acceptability criteria make the acceptability of a plan hard to define in general.

Reinforcement learning (RL) is a machine learning paradigm that trains agents to make sequential decisions in dynamic environments [2]. Agents learn to optimize their actions to achieve long-term objectives through trial and error guided by rewards or penalties. The decisions taken by dosimetrists when optimizing treatment can be formalized as an RL problem. Moreover, dosimetrists can guide the TPS towards an acceptable plan but usually struggle to explain their decision while interacting with the TPS. The difficulty in explaining why certain decisions are taken suggests using deep RL over expert-based methods. This setup is similar to image recognition, where one can say a picture represents a car or a boat but struggles to explain why.

The study's primary hypothesis is that all the information needed to decide what weights should be changed in the objective function used by the optimizer relies on the Dose Volume Histograms (DVHs). Our hypothesis is supported by the fact that dosimetrists almost solely use DVHs plots. In order to learn the actions of dosimetrists who use a TPS to optimize doses, we leverage deep learning. This is done by training an agent that takes the DVHs as the input of the current optimized dose, and predicts the evaluation of possible weights changes.

Typically, access to the exact actions taken by human dosimetrists on the TPS is unavailable (as clinics do not usually store this data; only the final plan is held). Therefore, we only use the dose distributions of previously treated patients to train our model. This partial availability of data suggests the use of RL.

2 Materials and Methods

We introduce a new paradigm for reward-based dosimetrist RL agents. This new reward system aims to improve how human-optimized doses are mimicked.

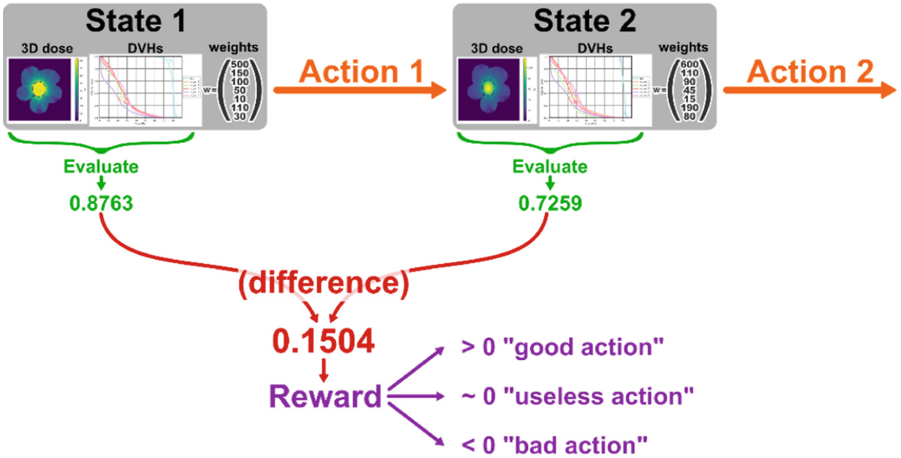


Fig. 1. Classical reinforcement learning reward for automatic dosimetry.

2.1 Reinforcement Learning Reward

In classical RL, we want $V(S_t) = R_t + \gamma V(S_{t+1})$ (so the update is $V(S_t) \leftarrow (1-\alpha)V(S_t) + \alpha[R_t + \gamma V(S_{t+1})]$). In the context of dose optimization, the reward R_t is defined as $R_t = \mathcal{E}(S_{t+1}) - \mathcal{E}(S_t)$, where $\mathcal{E}$ is a function that evaluates the quality of a state (such that higher is better; if lower is better, then swap S_t and S_{t+1}) (Fig. 1).

The evaluation $\mathcal{E}$ can be one or a mixture of the metrics mentioned in the introduction (Sect. 1) [9, 12, 13]. This setup may leverage knowledge about which actions to perform instead of guessing randomly, as a meta-optimizer would do. This could potentially gain some computation time.

However, this technique does not use past plans; it only needs the optimizer inputs (CT, structures contours). We propose using the availability of past treatment plans to more accurately reflect the complexity of decisions made by dosimetrists and better match their expectations of a fully automatic treatment planning system.

As developed in previous work, we can derive a distance between dose plans [11]. If we consider the clinical dose of past cases (used for training) as the best achievable one, we can evaluate a dose plan by computing its distance from the clinical dose plan.

Let D_t be the dose associated with S_t, and D_C the clinical dose. We then define $\mathcal{E}(S_t) = \mathcal{D}(D_t, D_C)$. Since, in that case, $\mathcal{E}(S_t)$ should be minimized, we will define the reward as

$$R_t = \mathcal{E}(S_t) - \mathcal{E}(S_{t+1}) = \mathcal{D}(D_t, D_C) - \mathcal{D}(D_{t+1}, D_C).$$

This reward can be interpreted as the “distance gained to the clinical dose”.

2.2 Architecture

We use a dense neural network, which inputs the DVHs and current normalized weight values. It outputs the $Q(s, a)$ value for each possible action a. Dense layers are very prone to overfitting. In order to force the network to actually predict the following evaluation for each possible action, without overfitting, we incorporated a bottleneck in the network (Fig. 2). Compressing the information stops the network from overfitting. Networks with such architecture show very little difference between training and validation sets (see Fig. 2).

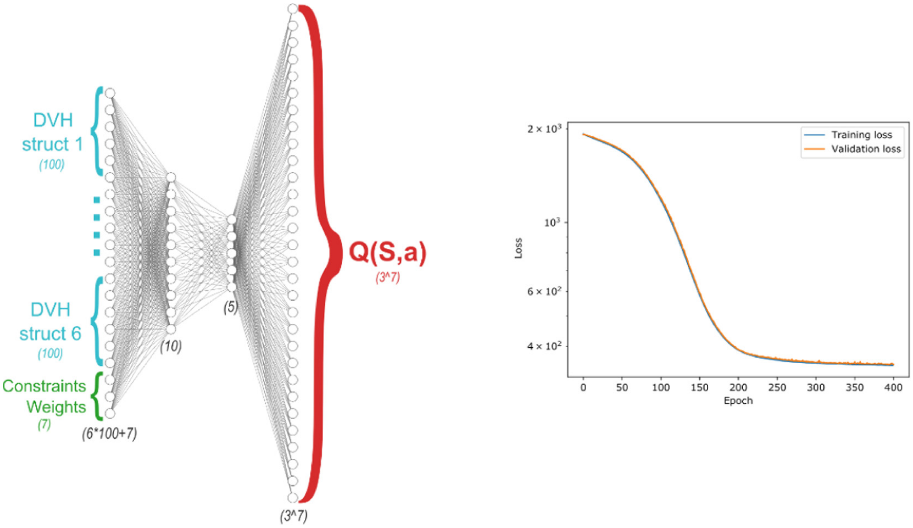


Fig. 2. Neural network architecture and loss evolution while training.

2.3 Avoiding Off-Distribution

We generated a training set of over 125k actions (this took five days on an NVIDIA GeForce GTX 1080). Despite this relatively large dataset, we have not explored exhaustively the state-actions space, and the network still lands off distribution. This can easily be spotted when the predicted Q value is greater than the current distance to the clinical dose; we choose to ignore those predictions, and in fact all outlier predictions. The justification is that our set of actions is limited, no action will suddenly drastically improve the plan. It is the combination of several sequential actions that allows good plan optimization. Therefore, while testing, we choose the action with the best prediction, while passing the outlier test just mentioned.

3 Results

Figure 3 shows how the distance between our RL agents performs over five steps on 30 test patients (unseen during the training). A lower distance is interpreted as an improved dose, since it is closer to the best dose, which is the clinical one.

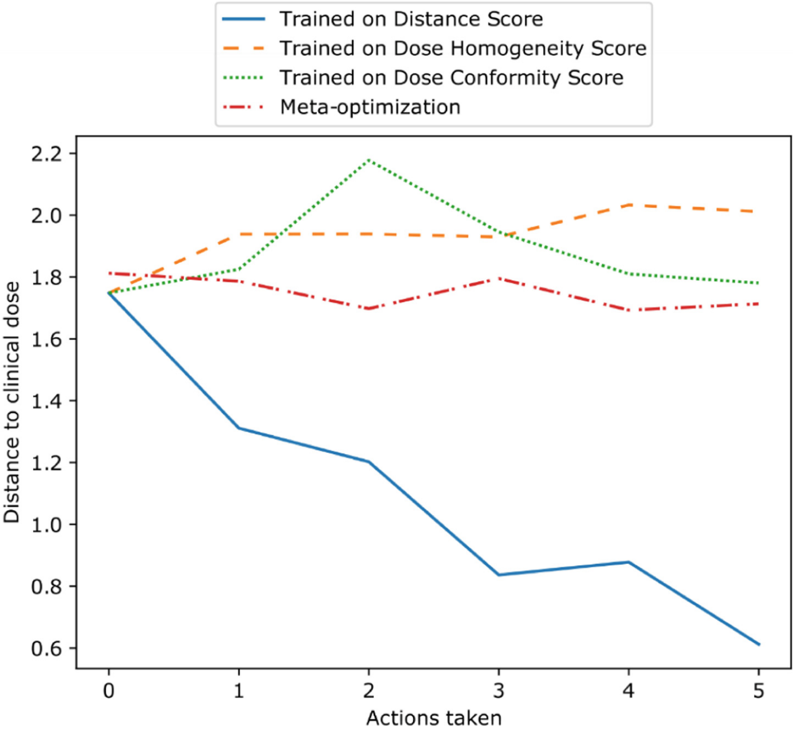


Fig. 3. Average distance between RL agent's dose and clinical dose.

3.1 Quantitative Results

The network converged on the training data, and validation showed minor overfitting. For testing, we generated 30 brand new cases that we again manually optimized. We then used the RL model to perform the optimization of these 30 unseen cases. On average, our model was able to reduce the dose distance with manually optimized dose by a factor of 3 (from 1.8 at iteration 0 to 0.6 at iteration 4), as shown in Table 1. We remark from the Table 1 that the homogeneity score and conformity score give similar results. Classical meta-optimization performs well, but needs a metric to elect the best dose (during the test, the clinical dose is unknown, so the DVHs distance metric is not available). We also observe that clinical doses are not always scoring high (in this test set, a high conformity, but low homogeneity compared to automatic techniques). This show the difficulty to create a metric that capture all the complexity of a clinically acceptable dose.

Table 1. Average performances of four algorithms tested on DVHs distance to clinical dose, dose homogeneity-based score, and conformity-based score.

Agent	Mean Final Distance[a]	Homogeneity Score[b]	Conformity Score[b]
RL on Distance Score	**0.612**	1.871	0.406
RL on Homogeneity Score	2.012	**4.387**	0.567
RL Conformity Score	1.770	4.017	0.507
Meta-optimization	N/A	4.117	**0.610**
Clinical doses	*0*	*1.541*	*0.580*

[a] *Distance is improved performance through a lower score.*
[b] *Score is improved performance through a higher score*

3.2 Qualitative Results

Figure 4 shows the DVHs at each of the first four optimization steps on one of the test patients, unseen by the agent during the training. Our model drastically reduced the dose distance with manually optimized doses. Visual inspection of the DVHs plot shows that the dose optimized by the RL agent is very close to the clinical (manually fine-tuned) one.

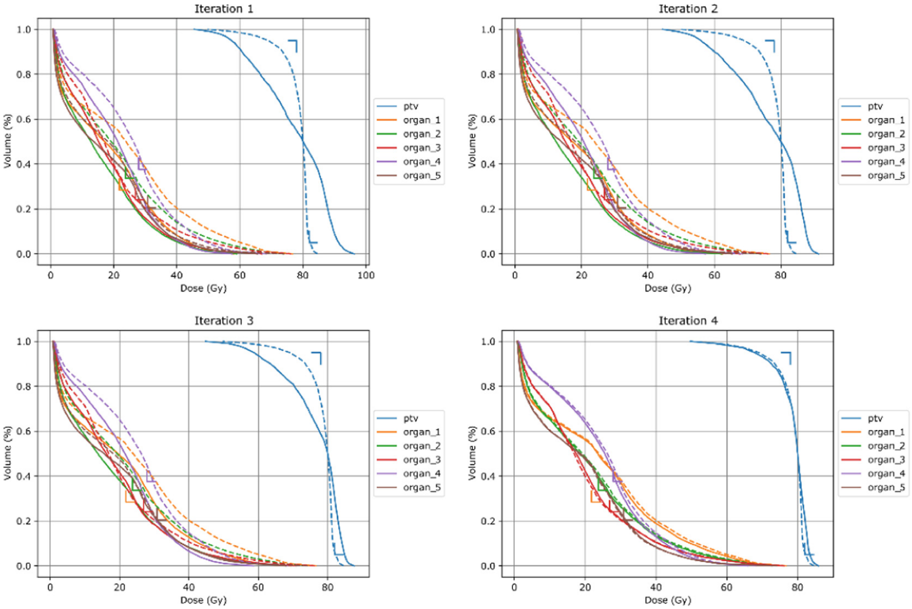


Fig. 4. RL Agent DVHs after each action taken on a test (unseen) patient. Solid lines are the agent's dose DVHs; dotted ones are the reference dose DVHs (manually finetuned).

4 Discussion

Our study demonstrates the potential of deep RL for automating radiotherapy treatment plan optimization. A key strength of our approach is its ability to learn from past treatment plans, capturing the complex decision-making processes of human dosimetrists. This data-driven approach avoids the limitations of predefined metrics, which may not fully capture the nuances of optimal treatment planning.

However, our study also has limitations. The agent's performance relies on the quality and quantity of available training data. Cases with limited historical data or complex anatomical features may require additional strategies. Moreover, while the agent achieves promising results regarding dose distance reduction, the dose is not guaranteed to be clinically acceptable. Although this study demonstrates the promise of our RL approach in a controlled setting, one final limitation to mention is that extending it to real-world radiotherapy planning would necessitates addressing additional complexities and constraints.

Several avenues exist for further research. Firstly, we plan to investigate strategies for incorporating additional information, such as patient characteristics and anatomical complexities, into the training process. Secondly, we aim to explore techniques for improving the interpretability of the agent's decision making process, allowing for better understanding and potential clinical validation.

5 Conclusion

Our approach differs from previous RL-based methods for radiotherapy planning in two key aspects. First, we avoid relying on pre-defined metrics for evaluation, which can be subjective, and limit the agent's ability to learn complex optimization strategies. Second, compared to traditional meta-optimization approaches, our method leverages past treatment data, potentially leading to more informed decision-making during the optimization process.

This study demonstrates deep RL's feasibility and potential benefits for automating radiotherapy treatment plan optimization. Our approach is capable of directly predicts state evaluations, and shows promise in achieving significant improvements in efficiency and, potentially, treatment outcomes. Further research is needed to address limitations, improve interpretability, and ensure safe clinical integration. This approach could revolutionize radiotherapy planning, leading to more standardized, efficient, and improved patient care.

Appendix

As this is very new and ongoing research, we generated synthetic phantom patients and associated trustable clinical doses. In future work, we hope to apply this technique to real cases.

Synthetic Phantom Patients

We generated 130 patients with oval axial section bodies. We set the body density to water density. We then added an ellipsoid PTV within the body, with a slightly different density (following $\mathcal{N}(1, 0.05)$). Likewise, we generate five organs gravitating around the PTV, aligned on the axial section (Fig. 5).

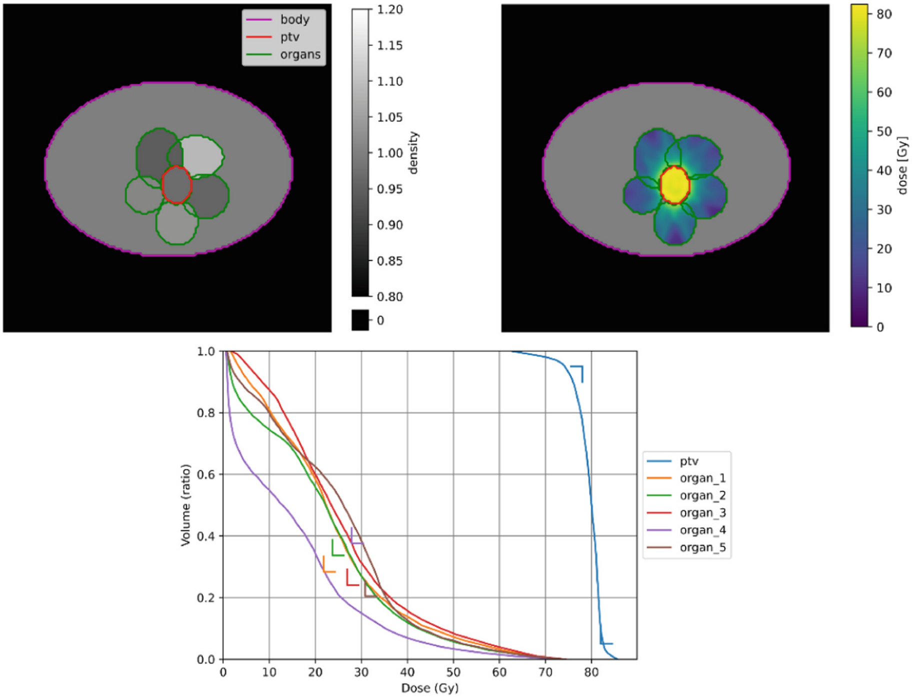


Fig. 5. Example of a (generated) patient:
Top-left: Main axial slice (center of the PTV) CT.
Top-right: Main axial slice (center of the PTV) of the clinical dose.
Bottom: Associated clinical dose DVH.

Clinical Dose

After generating the patient's CT and structures, we needed to create a reference dose that our agent should mimic. We manually set weights and performed a standard optimization. The dose prescription is a standard 80Gy on PTV, the same across all patients. We used a seven-beam IMRT irradiation technique on all the cohorts.

Optimization

We optimize the plan using the LBFGS optimizer (shown to be the most appropriate in [4]). For each DVH constraint (e.g. for PTV, D95 > 80 Gy), we used a linear penalization of the overdose.

References

1. Treatment planning system basics. Oncol. Med. Phys. https://oncologymedicalphysics.com/introduction-to-treatment-planning-systems/
2. Brooks, R.: What is reinforcement learning? December 2021. https://online.york.ac.uk/what-is-reinforcement-learning/
3. Das, I.J., Compton, J.J., Bajaj, A., Johnstone, P.A.: Intra- and inter-physician variability in target volume delineation in radiation therapy. J. Radiat. Res. (2021). https://doi.org/10.1093/jrr/rrab080. https://academic.oup.com/jrr/advance-article/doi/10.1093/jrr/rrab080/6367625
4. Dubois, P.: Radiotherapy dosimetry: a review on open-source optimizer, May 2023. http://arxiv.org/abs/2305.18014, arXiv:2305.18014 [cs, eess]
5. Galvin, J.M., Smith, A.R., Lally, B.: Characterization of a multileaf collimator system. Int. J. Radiat. Oncol. Biol. Phys. **25**(2), 181–192 (1993). https://doi.org/10.1016/0360-3016(93)90339-W. https://linkinghub.elsevier.com/retrieve/pii/036030169390339W
6. Huang, C., Yang, Y., Panjwani, N., Boyd, S., Xing, L.: Pareto optimal projection search (POPS): automated radiation therapy treatment planning by direct search of the Pareto surface. IEEE Trans. Biomed. Eng. **68**(10), 2907–2917 (2021). https://doi.org/10.1109/TBME.2021.3055822. https://ieeexplore.ieee.org/document/9343695/
7. Li, X., et al.: Input feature design and its impact on the performance of deep learning models for predicting fluence maps in intensity-modulated radiation therapy. Phys. Med. Biol. **67**(21), 215009 (2022). https://doi.org/10.1088/1361-6560/ac9882. https://iopscience.iop.org/article/10.1088/1361-6560/ac9882
8. Lyman, J.T.: Normal tissue complication probabilities: variable dose per fraction. Int. J. Radiat. Oncol. Biol. Phys. **22**(2), 247–250 (1992). https://doi.org/10.1016/0360-3016(92)90040-O. https://linkinghub.elsevier.com/retrieve/pii/036030169290040O
9. Moreau, G., François-Lavet, V., Desbordes, P., Macq, B.: Reinforcement learning for radiotherapy dose fractioning automation. Biomedicines **9**(2), 214 (2021). https://doi.org/10.3390/biomedicines9020214. https://www.mdpi.com/2227-9059/9/2/214
10. Oelfke, U., Bortfeld, T.: Inverse planning for photon and proton beams. Med. Dosim. **26**(2), 113–124 (2001). https://doi.org/10.1016/S0958-3947(01)00057-7. https://linkinghub.elsevier.com/retrieve/pii/S0958394701000577
11. Dubois, P., et al.: A Novel Framework for Multi-Objective Optimization and Robust Plan Selection Using Graph Theory, Glasgow, UK (2024)
12. Shen, C., Chen, L., Jia, X.: A hierarchical deep reinforcement learning framework for intelligent automatic treatment planning of prostate cancer intensity modulated radiation therapy. Phys. Med. Biol. **66**(13), 134002 (2021). https://doi.org/10.1088/1361-6560/ac09a2. https://iopscience.iop.org/article/10.1088/1361-6560/ac09a2
13. Shen, C., et al.: Intelligent inverse treatment planning via deep reinforcement learning, a proof-of-principle study in high dose-rate brachytherapy for cervical cancer. Phys. Med. Biol. **64**(11), 115013 (2019). https://doi.org/10.1088/1361-6560/ab18bf. arXiv:1811.10102 [physics]
14. Valentini, V., et al.: Survival after radiotherapy in gastric cancer: systematic review and meta-analysis. Radiother. Oncol. **92**(2), 176–183 (2009). https://doi.org/10.1016/j.radonc.2009.06.014. https://linkinghub.elsevier.com/retrieve/pii/S0167814009003247
15. Webb, S.: The physical basis of IMRT and inverse planning. Br. J. Radiol. **76**(910), 678–689 (2003). https://doi.org/10.1259/bjr/65676879. https://academic.oup.com/bjr/article/76/910/678-689/7470601
16. Wu, X., Zhu, Y.: An optimization method for importance factors and beam weights based on genetic algorithms for radiotherapy treatment planning. Phys. Med. Biol. **46**(4), 1085–1099 (2001). https://doi.org/10.1088/0031-9155/46/4/313. https://iopscience.iop.org/article/10.1088/0031-9155/46/4/313

17. Xing, L., Li, J.G., Donaldson, S., Le, Q.T., Boyer, A.L.: Optimization of importance factors in inverse planning. Phys. Med. Biol. **44**(10), 2525–2536 (1999). https://doi.org/10.1088/0031-9155/44/10/311. https://iopscience.iop.org/article/10.1088/0031-9155/44/10/311
18. Xu, D., Li, G., Li, H., Jia, F.: Comparison of IMRT versus 3D-CRT in the treatment of esophagus cancer: a systematic review and meta-analysis. Medicine **96**(31), e7685 (2017). https://doi.org/10.1097/MD.0000000000007685. https://journals.lww.com/00005792-201708040-0003

Reinforcement Learning with Balanced Clinical Reward for Sepsis Treatment

Zhilin Lu[1], Jingming Liu[2], Ruihong Luo[1], and Chunping Li[1](✉)

[1] School of Software, Tsinghua University, Beijing 100084, China
cli@tsinghua.edu.cn
[2] Beijing Tiantan Hospital, Capital Medical University, Beijing 100070, China

Abstract. Sepsis, a severe reaction to infection, presents significant challenges in intensive care units (ICUs), often resulting in high mortality rates. Traditional treatment approaches, primarily reliant on clinicians' judgment and standard guidelines, frequently fail to deliver personalized care. Moreover, clinical decisions may vary considerably among healthcare providers managing identical patient cases. In this study, we propose an innovative method for optimizing sepsis treatment strategies through Deep Reinforcement Learning (DRL), leveraging patient data, medical expertise, and comprehensive sepsis research. Additionally, we develop an interpretable reward formulation to guide the DRL agent in learning from real clinical data, aiming to enhance treatment outcomes and mitigate mortality risks. Our results demonstrate that the DRL approach surpasses existing methods, leading to safer sepsis treatment decisions and correlating with increased patient survival rates. This investigation underscores the potential of Artificial Intelligence (AI) in enhancing treatments for sepsis and other intricate medical conditions.

Keywords: Sepsis treatment · Optimal policy · Reward function · Reinforcement learning · Clinical decision support systems (CDSSs)

1 Introduction

Sepsis, a life-threatening infection response, presents a major challenge in ICUs globally, with high mortality rates [1]. It contributes to about 20% of global deaths, varying by severity [2]. The variability in clinical decisions and the pressing need for improved management strategies emphasize the importance of advancing sepsis treatment. Traditional sepsis treatment emphasizes hemodynamic management and blood pressure stabilization, but optimizing intravenous fluid volumes and vasopressor dosages remains complex, often leading to suboptimal outcomes [3–6]. In 2016, Komorowski et al. pioneered the use of reinforcement learning (RL) to optimise the dosage combination of intravenous fluids and vasopressors in treating sepsis patients, setting a precedent for future research in this domain [7]. They employed terminal reward at the end of each trajectory, contingent on 90-day clinical outcome (life or death). And they used value iteration techniques to discover the optimal policy. However, relying on 90-day clinical outcome as a reward did not effectively differentiate between patients with varying survival lengths,

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 161–171, 2024.
https://doi.org/10.1007/978-3-031-66538-7_17

and the evaluation based on Q-values provided by the network lacked persuasive power. In a recent reproducible study on the dosage combination of intravenous fluids and vasopressors, Wu et al. utilized reward functions based on the Sequential Organ Failure Assessment (SOFA) scores and 90-day clinical outcomes, concluding that patient survival rates are highest when the model achieves the maximum expected return [8]. This finding underscores the model's sensitivity to patient survival rates, although it does not guarantee high survival rates under all circumstances. Therefore, beyond reward functions, there is a need to explore further in terms of evaluation methods to establish more objective and convincing standards.

To date, related studies have not thoroughly investigated reward functions nor analyzed their impact on model decisions. Furthermore, evaluation methods still require improvement. Hence, we embark on research into reward functions, leveraging offline reinforcement learning algorithms proven for their stability in this research domain, and proposing more persuasive evaluation methods.

Specifically, we begin with the logic of drug effects, exploring the impact of indicators influenced by medication on the human body throughout the sepsis progression, thus resulting in a more interpretable reward function. We further propose a new evaluation method. By grouping patients based on strategy distance and comparing survival curves across different SOFA scores segments and treatment groups, we demonstrate that our approach can effectively identify inappropriate clinician decisions. Patients whose treatment trajectories were closer to our agent's decisions indeed experienced higher survival rates.

2 Methods

2.1 Reinforcement Learning Framework for Sepsis Treatment

Utilizing Komorowski's data extraction scheme [7] and the Medical Information Mart for Intensive Care (MIMIC)-III v1.4 database [9], we access a dataset of ICU patient records to analyze sepsis treatment. Illuminated by a methodology aligned with Wu et al. [8], our model adopts a continuous state space derived from patient vital signs over 4-h intervals and a discrete 25-action space, ranging from (0,0) to (4,4), for combination of vasopressors and intravenous fluids. To facilitate comparison, we also employs the strategy of WD3QN method [8], using the same network architecture, that leverages an adaptive dynamic weight p to achieve a trade-off between Dueling DQN and D3QN, thereby deriving the optimal policy. The Q-value calculation formulas are shown in Eqs. (1) and (2).

$$Q(S_t, a_t) = p \times \max_{a_t} Q\big(S_t, a_t; \omega^-\big) + (1 - p) \times Q\big(S_t, \operatorname{argmax}_{a_t} Q(S_t, a_t; \omega); \omega^-\big) \quad (1)$$

$$Q(S_{t-1}, a_{t-1}) = r + \gamma Q(S_t, a_t) \quad (2)$$

The Q-function, $Q(S, a)$, represents the expected return for a state-action pair, updated by weighting contributions from the main ω and target ω^- networks using p. r is the reward for an action at state S_{t-1}, and γ is the discount factor. We primarily focus on the configuration of the reward function.

2.2 Reward Formulation

In this study, we scrutinize the reward function previously reliant on the SOFA score. Although SOFA correlates with mortality, using it for immediate rewards in reinforcement learning may not be entirely suitable. We observed that agents tend to over-administer vasopressors when SOFA scores guide rewards. Our analysis investigates changes in organ-specific SOFA scores after vasopressor administration. Changes in total SOFA score are highly correlated with changes in circulatory SOFA score, as shown in Fig. 1 and Table 1.

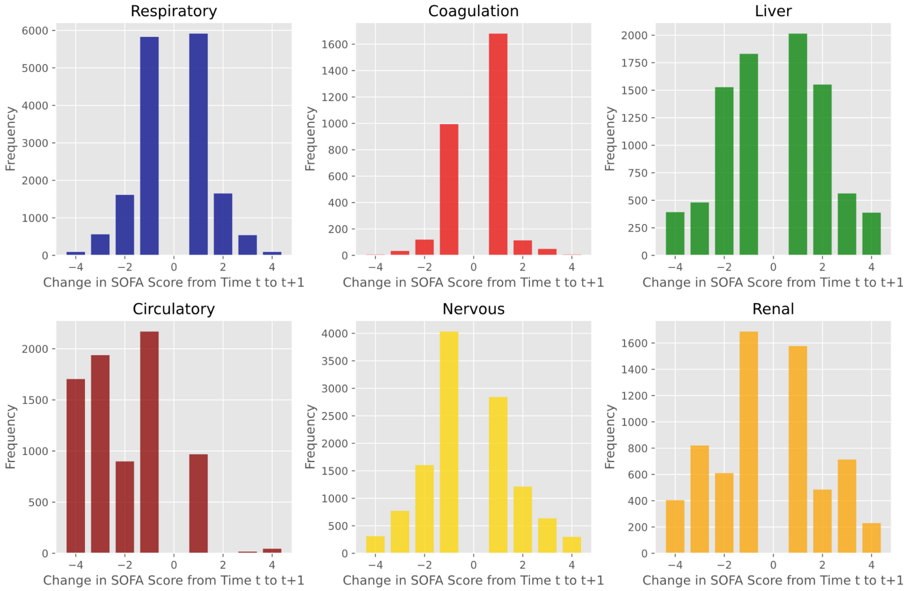


Fig. 1. Histograms of SOFA score changes across six systems. In each system, the most frequent change between adjacent time points is ‘0’. However, to better illustrate the statistics of the changed values, ‘0’ is not included in the diagrams.

Table 1. Slope and bias obtained from linear regression of SOFA changes for each system against the total SOFA changes. “Correlation” denotes the correlation coefficient.

Parameter	Respiratory	Coagulation	Liver	Circulatory	Nervous	Renal
Slope	1.0047	1.2042	1.0410	1.0959	1.1110	1.0224
Bias	−0.0700	−0.0729	−0.0714	**−0.0041**	−0.0591	−0.0641
Correlation	0.4186	0.1741	0.4364	**0.5341**	0.4807	0.3906

It is noteworthy that the calculation method for the circulatory SOFA takes into account the use of vasopressors (see Table 2 in appendix). That is, when vasopressors are administered, the circulatory SOFA is elevated, and once the patient's blood pressure increases and medication is stopped, the circulatory SOFA decreases. This suggests that while vasopressors are effective in enhancing circulation and thereby reducing the total SOFA score, they may lead to a tendency within the agent to favor higher dosages due to their immediate beneficial effects, overlooking potential adverse effects. Consequently, although the total SOFA score serves as a credible indicator of patient health status, its application as a reward in reinforcement learning algorithms requires adjustment to prevent the encouragement of excessive vasopressor administration. To formulate the immediate reward in more details for clinical reinforcement learning, we analyze vasopressor and fluid treatment in sepsis, consulting a wide range of medical literature to understand key principles and their interplay, which constructs the causal hierarchy diagram for sepsis treatment, shown as Fig. 2.

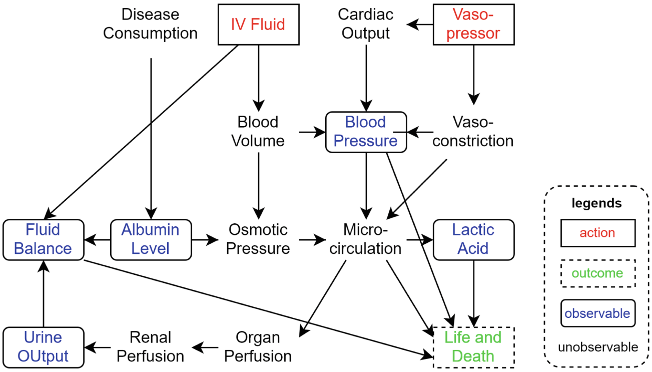


Fig. 2. Causal hierarchy diagram for sepsis treatment.

The causal relationships between the nodes depicted in Fig. 2 are supported by the following medical standpoints [10, 11]. **Intravenous Fluids**: Fluid resuscitation and management are crucial for maintaining blood volume and homeostasis, affecting solute distribution in vascular and extravascular spaces by altering blood volume and osmotic pressure. **Vasopressors**: By inducing vasoconstriction, vasopressors raise blood pressure, affecting microcirculation and organ perfusion, and modulate cardiac output. **Fluid Balance and Osmotic Pressure**: Body fluid equilibrium influences solute distribution, impacting tissue perfusion and oxygenation, which affects lactate production and clearance. Elevated osmotic pressure can shift fluids out of cells, with albumin playing a key role in fluid distribution and beyond. **Microcirculation**: Essential for tissue oxygenation and nutrient delivery, microcirculation is influenced by osmotic pressure, vasoconstriction, cardiac output and blood pressure, crucial for organ perfusion, especially renal function. **Life and Death**: Outcomes are shaped by fluid balance, arterial pressure, microcirculation, and lactate levels.

According to the observable variables in Fig. 2, we formulate the immediate reward, focusing on blood pressure, fluid balance, albumin, lactate, excluding urine output as it is already encapsulated within fluid balance. A linear reward function, $R_{\text{immediate}}$, considering the changes over adjacent four-hour intervals in mean arterial pressure ΔMAP, fluid balance ΔFB, albumin ΔALB, and lactate levels ΔLAC for simplicity is shown in Eq. 3.

$$R_{\text{immediate}} = \alpha \Delta \text{MAP} + \beta \Delta \text{FB} + \gamma \Delta \text{ALB} + \delta \Delta \text{LAC} \tag{3}$$

Due to mean arterial pressure and fluid balance being the most densely connected observable nodes and closely linked to action nodes, they are directly influenced by therapeutic actions. Conversely, albumin and lactate, situated further from direct therapeutic interventions and affected by numerous factors, will be assigned a reduced weight in the reward function. Following the principles of classical reinforcement learning [12], we target an immediate reward range between -1 and 1 for reinforcement learning algorithms. For mean arterial pressure and fluid balance, we determine their coefficients in such a manner that the coefficient multiplied by the absolute value of the median change observed in the MIMIC-III dataset over every four-hour interval approximately equals 1. For albumin and lactate, we adjust their coefficients so that the product of the coefficient and the absolute value of their median change aligns closely with 0.1.

We further formulate a continuous terminal reward, R_{terminal}, varying linearly with survival days for nuanced feedback and optimizing for long-term patient survival, not merely limited in a 90-day cutoff. Patients who survive for 90 days or more will receive a constant reward of R. This refined approach is aimed at providing a more comprehensive and detailed evaluation of the treatment outcomes, ensuring that the evaluation process captures a broader range of patient experiences and outcomes.

$$R_{\text{terminal}} = \begin{cases} R/45 \times survive\_days - R, & when\, survive\_days < 90 \\ R, & when\, survive\_days \geq 90 \end{cases} \tag{4}$$

Equations 3 and 4 as the Balanced Clinical Reward explicitly aim to strategically guide the agent towards an optimal equilibrium between immediate treatment effects and long-term health outcomes.

3 Experiments

Integrating findings from a recent pivotal study in this field, we have selected the recent work [8] as our benchmark. A 5-fold cross-validation method was applied, with outcomes averaged after training both models over 100 epochs each.

3.1 Action Distribution

Given the application of reinforcement learning algorithms in the clinical decision supporting for sepsis treatment, where their real-world efficacy remains unverified, analyzing the action distribution offers crucial insights into the algorithmic decision-making preferences. The aggregate action distribution of physician and two agents across the test set at each time point, derived from averaging five experimental outcomes, is illustrated in Fig. 3.

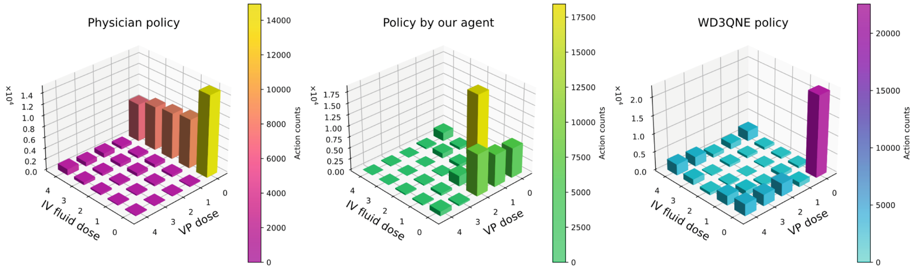


Fig. 3. Action distribution of physician policy (left), policy by our agent (center) and WD3QNE[8] policy (right). The height of the bars represents the frequency of actions.

The analysis indicates significant differences in how actions are chosen, largely due to different reward systems. When changes in total SOFA scores are utilized as immediate rewards and clinical outcomes serve as terminal rewards, we see a unique pattern in the agent's decisions: a decrease in moderate actions with an increase in either minimal or maximal interventions. This suggests a tendency towards extreme strategies—either minimal use or maximal application of treatment options. This pattern aligns with previous concerns about using SOFA score changes for immediate rewards, leading to an emphasis on short-term gains without fully considering long-term risks.

However, under the setting of Balanced Clinical Reward, the agent tends towards a more moderate approach. The action distribution primarily shows a peak for moderate strategies and decreases for extreme actions. This suggests that the Balanced Clinical Reward encourages the agent to find a middle ground between immediate advantages and potential long-term disadvantages.

3.2 90-Day Mortality Rate

While action distribution offers insights into the overall tendencies in decision-making, it lacks direct clinical applicability.

A more empirically grounded method of assessment is warranted. We recognize the limitations inherent in previously suggested methodologies, such as comparing model strategies to those of physicians using Q values [13] or observing the relationship between expected returns and mortality rates [8]. These methods fall short of objectivity. The comparison of model strategies against physician decisions through Q values inherently skews towards the algorithmic decisions, given that reinforcement learning algorithms prioritize actions with the highest Q values. Consequently, the model invariably positions the physician's decisions at a Q value that is not higher than the agent's chosen action, undermining objective assessment. And it's acknowledged that Q values derived from reinforcement learning models are heavily influenced by the reward function. With an emphasis on the terminal outcome of life or death in the traditional reward function, the model's Q values predominantly reflect the likelihood of mortality. However, this does not guarantee that the model can consistently achieve high Q values with its proposed strategies across all states. Therefore, comparing mortality rates across models based merely on reward intervals highlights the varied sensitivities of models towards patient mortality rather than the efficacy of decisions. A more robust agent may accurately depict an individual's survival probability through expected returns, yet this does not validate the utility of the agent's decisions. Our focus should pivot towards making the most effective decisions by examining comprehensive data from the entire patient cohort.

Komorowski et al. explored the association between mortality and dosage excess, where dosage excess denotes the deviation between administered and recommended dosages of intravenous fluids and vasopressors, averaged across all patient time points [13]. However, the quintessential clinical challenge in sepsis decision-making is optimizing the balance between vasopressor and fluid dosages for an improved therapeutic outcome. Consequently, we introduce a novel comparative approach that amalgamates both strategies rather than isolating the examination to the disparity and mortality rate between physician and algorithmic strategies within a singular framework. The distance between two decision strategies for intravenous fluids and vasopressors at time t, given their respective actions ranging from 0 to 4, can be mathematically represented using the Euclidean distance formula. For two strategies A and B, with A represented by the action pair $a_{\text{IV}}, a_{\text{VP}}$ and B by $b_{\text{IV}}, b_{\text{VP}}$, where a_{IV} and a_{VP} are the actions for intravenous fluids and vasopressors for strategy A, and b_{IV} and b_{VP} are the actions for strategy B, the formula for the strategy distance is shown as Eq. 5.

$$D^t_{\text{A - B}} = \sqrt{(a_{\text{IV}} - b_{\text{IV}})^2 + (a_{\text{VP}} - b_{\text{VP}})^2} \tag{5}$$

This definition interprets the distance between different action combinations on the action distribution graph's xOy plane as the Euclidean distance between two points representing actions. It facilitates a comprehensive strategy comparison, aligning with the objective of addressing sepsis clinical decision-making challenges. With this distance definition, the correlation between the algorithm's decision distance to physician choices and patient mortality is shown in Fig. 4.

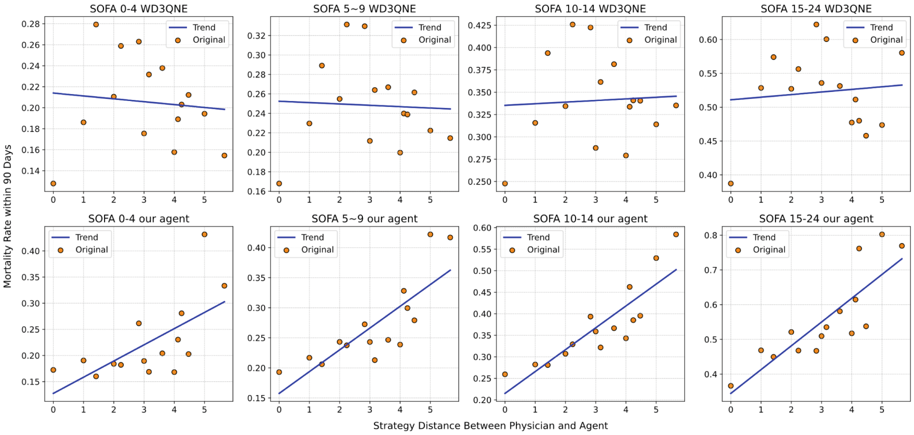


Fig. 4. Relationship between the 90-day mortality rate and the strategy distance between doctors and algorithms per patient at each time point. The dots represent the original data, and the line represents the trend line obtained by linear fitting the data.

Figure 4 illustrates the linkage between mortality rates and strategy distances across distinct SOFA score categories. Within the framework of the Balanced Clinical Reward, a consistent pattern is observed across all SOFA segments: the greater the divergence between the physician's and the algorithm's decisions, the higher the mortality rate observed. Conversely, under a reward configuration emphasizing clinical outcomes and SOFA scores, a discernible trend correlating strategy distance with mortality rates was not evident.

Nonetheless, relying solely on a 90-day mortality-strategy distance chart does not adequately address the disconnect between the decisional discrepancies of physicians and algorithms and the patients' overall health trajectory. Matters of life and death, while instantaneous, are the culmination of a series of decision-making processes. The 90-day mortality-strategy distance chart, which treats each point in the patient's treatment journey as an independent sample, fails to capture the temporal aspect of these decisions. What is necessitated is an analysis that incorporates a four-dimensional event model: strategy distance, temporal progression, the patient's initial SOFA score at sepsis diagnosis, and the eventual outcome of life or death. This realization prompts the inclusion of temporality as a critical dimension for evaluation, leading to the adoption of survival curves for a more systematic and comprehensive analysis.

3.3 90-Day Survival Curves

To conduct time-series analysis, we defined the average strategy distance between the algorithm and doctors throughout the entire treatment process as Eq. 6.

$$\text{Average Strategy Distance} = \sqrt{\sum_{t=1}^{T}\left(D^{t}_{\text{alg - doc}}\right)^{2}/T} \tag{6}$$

Considering treating strategy as a vector, the average distance between two sequential strategies still utilizes the Euclidean distance calculation method. Dividing by T inside the square root ensures that the average strategy distance can maintain the same range as the strategy distance at a single time point.

A survival curve allows observing a patient group's survival status from two dimensions: time and life/death. Therefore, to conduct an objective survival analysis, we need to segment the other two dimensions (average strategy distance and initial patient SOFA) to define patient groups. For initial SOFA, we divided it into four segments. For average strategy distance, we used half the maximum distance as the boundary shown in Fig. 5.

It's apparent that, under the Balanced Clinical Reward setting, there are significant differences between survival curves of different patient groups within the SOFA 5–9 and SOFA 10–14 segments, aligning with findings in previous research. Patients within the SOFA 0–4 segment, having relatively good conditions, are insensitive to treatment strategies, showing no significant differences in survival curves. Patients within the SOFA 4–14 segment are in a transitional state where treatment can affect their outcomes, showing sensitivity to different treatment strategies. For patients in the SOFA 15–24 segment with poor conditions, different treatment strategies do not significantly diverge the survival curves.

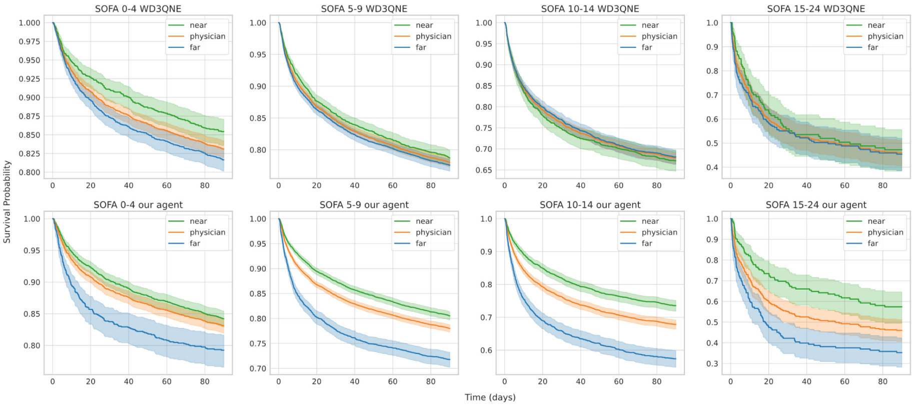


Fig. 5. Kaplan-Meier survival curves illustrate the outcomes for patient groups under different treatment strategies. Each subplot corresponds to a specific initial SOFA segment. Three survival curves are depicted for each segment: one for patients with an average strategy distance from doctors to algorithms less than $2\sqrt{2}$, one for distances greater than $2\sqrt{2}$, and one representing the overall patient survival under doctor decisions. The shaded areas around the curves indicate the 95% confidence intervals. Statistically significant differences between the “near” and “far” survival curves, as identified by our agent, were observed across all SOFA segments.

3.4 External Validation

It was noted that in the MIMIC-III dataset, doctors tend to use a significant amount of fluid administration. This might be a reason why fluid balance shows outstanding results in the Balanced Clinical Reward. To validate the universality of the reward, we extracted treatment records of sepsis patients from the eICU database, selecting 3200 records with a low missing rate. We conducted external validation on the records using an agent trained on MIMIC-III dataset. The survival curves for external validation are illustrated in Fig. 6.

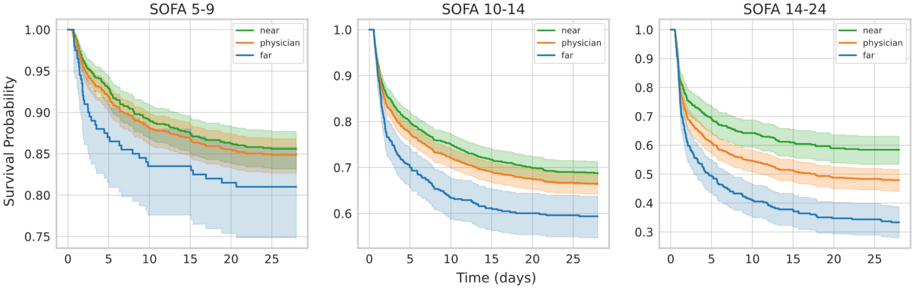


Fig. 6. Kaplan-Meier survival curves resulting from external validation on the eICU dataset. In the 3200 patient records we extracted, with low missing data rates (50%), there were only 20 patients in the initial SOFA 0–4 segment, which does not have significant statistical meaning; hence, it is not depicted in the graph. External validation demonstrates that the reward function proposed in this paper is generalizable across different datasets.

4 Conclusion

In this paper, we propose a novel method for optimizing sepsis treatment strategies based on Deep Reinforcement Learning, and we further design an interpretable reward function that guides the DRL agent in learning from real clinical data to improve treatment outcomes and reduce mortality risks. Our findings show that our DRL agent outperforms existing methods in making safer sepsis treatment decisions and is linked to higher patient survival rates, demonstrating the potential of reinforcement learning in healthcare. This work marks a significant step towards the interpretability and practical application of reinforcement learning in clinical decision-making for sepsis. We have provided a valuable methodological foundation for applying this technology in other clinical scenarios. Through detailed analysis and empirical validation, we have showcased the immense potential of reinforcement learning to enhance treatment outcomes, paving the way for further research and practice. Future endeavours will integrate mathematical models from sepsis pathology for more accurate modeling, aiming to improve survival curves.

Appendix

Calculation method of SOFA Score for Cardiovascular System is shown in Table 2.

Table 2. SOFA Score for the Cardiovascular System. Drug abbreviations: Dop = Dopamine, Dob = Dobutamine, Epi = Epinephrine, Nor = Norepinephrine.

Score	Mean arterial pressure/Administration of vasopressors required
0	MAP >= 70 mmHg
1	MAP < 70 mmHg
2	Dop <= 5 mcg/kg/min or Dob (any dose)
3	Dop > 5 mcg/kg/min, Epi <= 0.1 mcg/kg/min, or Nor <= 0.1 mcg/kg/min
4	Dop > 15 mcg/kg/min, Epi > 0.1 mcg/kg/min, or Nor > 0.1 mcg/kg/min

We conducted an ablation study, with results shown in Table 3.

Table 3. 90-day survival rate results of ablation study. The abbreviations MAP, ALB, FB, LAC, and Terminal represent the scenarios without each item.

SOFA	Patients	MAP	ALB	FB	LAC	Terminal	All
	near	83.95%	83.76%	83.99%	84.29%	83.83%	84.22%
0–4	phys	83.11%	83.11%	83.11%	83.11%	83.11%	83.11%
	far	80.87%	80.05%	82.70%	79.12%	80.55%	79.24%
	near	79.85%	80.35%	75.78%	80.29%	80.03%	80.52%

(*continued*)

Table 3. (*continued*)

SOFA	Patients	MAP	ALB	FB	LAC	Terminal	All
5–9	phys	78.00%	78.00%	78.00%	78.00%	78.00%	78.00%
	far	73.49%	71.05%	78.60%	72.14%	72.67%	71.76%
	near	72.24%	72.85%	64.44%	72.99%	73.10%	73.53%
10–14	phys	67.84%	67.84%	67.84%	67.84%	67.84%	67.84%
	far	59.93%	57.45%	69.04%	58.30%	57.63%	57.37%
	near	52.05%	55.43%	44.07%	55.49%	54.60%	57.41%
15–24	phys	45.99%	45.99%	45.99%	45.99%	45.99%	45.99%
	far	39.39%	35.85%	47.00%	36.84%	36.71%	35.26%

References

1. Singer, M., Deutschman, C.S.: The third international consensus definitions for sepsis and septic shock (Sepsis-3). JAMA **315**(8), 801–810 (2016)
2. Rudd, K.E., Johnson, S.C.: Global, regional, and national sepsis incidence and mortality, 1990–2017. Lancet **395**(10219), 200–211 (2020)
3. Vincent, J.-L., Lefrant, J.-Y.: Comparison of European ICU patients in 2012 (ICON) versus 2002 (SOAP). Intensive Care Med. **44**, 337–344 (2018)
4. Malbrain, M.L.N.G.: Fluid overload, de-resuscitation, and outcomes in critically ill or injured patients. Anaesthesiology Intensive Therapy **46**(5), 361–380 (2014)
5. Marik, P.E., Linde-Zwirble, W.T.: Fluid administration in severe sepsis and septic shock, patterns and outcomes. Intensive Care Med. **43**, 625–632 (2017)
6. Waechter, J., Kumar, A.: Interaction between fluids and vasoactive agents on mortality in septic shock. Crit. Care Med. **42**(10), 2158–2168 (2014)
7. Komorowski, M., Gordon, A., Celi, L.A., Faisal, A.: A Markov decision process to suggest optimal treatment of severe infections in intensive care. In: NIPS Workshop on ML for Health (2016)
8. Wu, X., Li, R.: A value-based deep reinforcement learning model with human expertise in optimal treatment of sepsis. NPJ Digit. Med. **6**(1), 15 (2023)
9. Johnson, A.E.W.: MIMIC-III, a freely accessible critical care database. Sci. Data **3**(1), 1–9 (2016)
10. Sakr, Y., Bauer, M.: Randomized controlled multicentre study of albumin replacement therapy in septic shock (ARISS). Trials **21**, 1–13 (2020)
11. Lee, J., de Louw, E.: Association between fluid balance and survival in critically ill patients. J. Intern. Med. **277**(4), 468–477 (2015)
12. Rao, R.P.N.: Reinforcement Learning: An Introduction. In: Sutton, R.S., Barto, A.G. (eds.) vol. 1998, 380 pages. MIT Press, Cambridge (2000). ISBN 0-262-19398-1
13. Komorowski, M.: The artificial intelligence clinician learns optimal treatment strategies for sepsis in intensive care. Nat. Med. **24**(11), 1716–1720 (2018)

Secure and Private Vertical Federated Learning for Predicting Personalized CVA Outcomes

Corinne G. Allaart[1,2,6](✉), Marc X. Makkes[1,3,6], Lea Dijksman[2,6],
Paul van der Nat[2,5,6], Douwe Biesma[4,6], Henri Bal[1,6],
and Aart van Halteren[1,3,6]

[1] Vrije Universiteit, Amsterdam, The Netherlands
c.g.allaart@vu.nl
[2] St. Antonius Hospital, Nieuwegein, The Netherlands
[3] Fortaegis Technologies, Amsterdam, The Netherlands
[4] Philips, Eindhoven, The Netherlands
[5] Leiden University Medical Center, Leiden, The Netherlands
[6] IQ Healthcare, Radboud UMC, Nijmegen, The Netherlands

Abstract. Cerebrovascular accident (CVA) outcome predictions could improve patient-centered care by informing individual patients on rehabilitation and expected outcomes. However, CVA patients' data is vertically distributed across hospitals and rehabilitation clinics. Centralizing distributed medical data in a central repository leads to difficulty concerning data privacy and data ownership. Vertical federated learning has been introduced as a solution, but it is not secure. We introduce our secure vertical federated learning (SVFL) protocol that prevents label and data leakage through encrypted active-party backpropagation. We use this to produce the first CVA outcome model using hospital and rehabilitation data in a vertically federated setting. Data from 825 CVA patients admitted to the St. Antonius Hospital, the Netherlands was collected, including their rehabilitation trajectory in three clinics, to predict functional status (dichotomized mRS score) after 3 months. Our results show that a model trained on the vertically integrated hospital and rehabilitation data performs better than a model trained on either of these sets alone. Training using SVFL yields a slightly lower predictive performance compared to training on a fully centralized data set. No difference in predictive performance between secure and unsecured VFL was observed, although secure VFL is computationally more expensive. This highlights that SVFL is a promising alternative for situations where it is not possible (or desired) to centralize vertically partitioned data.

1 Introduction

Like for many other diseases, patients with a cerebrovascular accident (CVA) have a long aftercare process after hospital treatment [fDCP23]. The first 3 months after a CVA can have a great influence on the eventual recovery.

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 172–181, 2024.
https://doi.org/10.1007/978-3-031-66538-7_18

To deliver patient-centered care and make informed choices on rehabilitation, it is important to inform individual patients about the right rehabilitation and expected outcomes. A personalized outcome prediction is an essential component of this. Using readily available clinical data, such as data collected for evaluation and comparing of care initiatives (as in value-based health care [PT06]), can lead to quicker adoption of prediction models [BIJC+20]. Previously, we have shown that even with a small dataset, some deep learning (DL) models can have better predictive performance than standard ML algorithms like random forests and gradient boost classifiers [BAvH24]. [HLJ+21] showed that incorporating patient data collected after discharge can lead to better predictions. However, as CVA patients are usually treated sequentially in multiple healthcare institutions, their data is distributed across these institutions such as the hospital and rehabilitation clinics. Centralizing distributed medical data through a central repository leads to difficulty both in data privacy and in data ownership, for example leading to issues with shared data structuring, analysis, and matching of patient records. In this paper, we examine how vertical federated learning can be used to overcome these challenges securely.

There are two different types of federated learning (FL): Horizontal FL focuses on scenarios where the data samples are different but share the same features (see Fig. 1a), while vertical FL (VFL) addresses scenarios where parties possess different features but share data samples (Fig. 1b). In the case of patients being managed by multiple care institutions, this refers to vertically partitioned data. Recent developments in VFL have opened up the possibility of developing AI algorithms without bringing data into a central repository. A common way to perform vertical federated learning is through vertically split networks [CSM+20], which consist of a neural network split among the different parties (Fig. 2). Multiple (passive) parties have a subset of the features, and the central or active party has the labels. The passive parties calculate the bottom layers of the network, and the active party uses the representations to train the top layers of the models. As such, all parties retain ownership of their data and their part of the network.

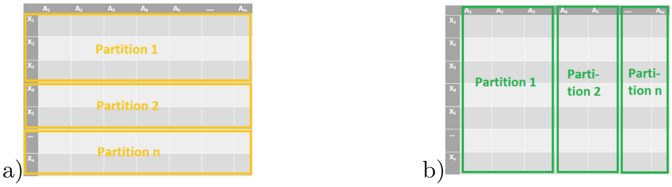


Fig. 1. Shows (a) Horizontally partitioned data and (b) Vertically partitioned data, with rows representing samples or patients and columns representing features

There are several issues with VFL. It can lead to lower predictive performance [AKBVH22], but most importantly, it is not secure: it can lead to data leakage, where parties can recover input data of the other party, and label leakage, where

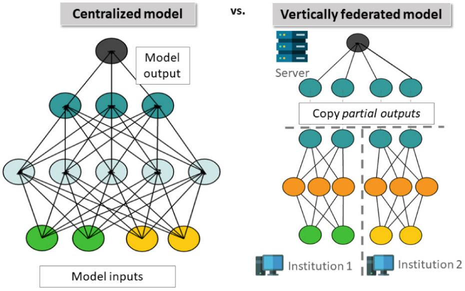


Fig. 2. Vertical Federated Learning with split networks. Left shows a centralized neural network, with right the vertically federated version distributed over two institutions and a central party.

the party without the label can recover the labels of the patients. Some papers use differential privacy to counteract these security issues [LWXO21], but this leads to a loss in predictive performance [JCH+21] and might not prevent label leakage. Another option is encryption to ensure security for example through a two-party system [FXC+22] or by secret sharing the cut layer [YHL+22], but this can still be vulnerable to attacks [NHDC23]. We aim to provide an improvement of the current methods of VFL by providing security against label leakage. This is achieved by letting the party with the labels perform the backpropagation in its entirety, but have partial models and datasets be encrypted with a cryptographic function with homomorphic properties. Thus, no information can be extracted during backpropagation. Moreover, we apply this secure VFL to provide an analysis of the use of vertically partitioned data for CVA outcome predictions. For this purpose, we have collected a federated dataset on CVA rehabilitation from the St. Antonius Hospital and three adjacent rehabilitation clinics, to predict the functional outcome three months after a CVA. As far as we know, this will be the first vertical federated learning model for CVA outcome prediction. We will also provide evaluations of two public stroke datasets for a more robust analysis. As such, our contributions are as follows:

- A secure vertical federated learning protocol that prevents label and data leakage through encrypted active-party backpropagation
- The first CVA outcome model using hospital and rehabilitation data in a vertically federated setting
- Evaluation of secure vertical federated learning on predictive performance, and its trade-offs with security, privacy, and efficiency using multiple clinical datasets

2 Proposed Protocol

Consider a situation where the hospital and the rehabilitation clinics want to cooperatively train a VFL network, where the hospital has part of the data and the labels, and as such is the active party. The rehabilitation clinics (passive parties) have data to contribute. A *trusted third party* (TTP) is present to distribute and encrypt the models and train the top layers of the model. Cryptographic functions with homomorphic properties enable computations directly on encrypted data, maintaining confidentiality throughout the process, such as encrypted backpropagation. After decryption, the results are the same as if the computations were performed on the plaintext data. This allows us to set up an SVFL with encrypted partial models and data. There are several constraints to ensure security: To prevent label leakage, *(1) Only the hospital has access to the labels.* To prevent data leakage, *(2) The hospital and rehabilitation clinic are not allowed to share their data. (3) Parties with data or labels have no access to the full model, only in encrypted form*, so the TTP does not share the (combined) model in unencrypted form. *(4) The hospital and rehabilitation clinic only send their partial outputs and partial models to the TTP* so the TTP cannot extract any information. As such, we create a protocol where the data and models are encrypted by the TTP public key, all parties train their partial model locally, and the active party performs the encrypted backpropagation and uses the result for forward propagation. The functions detailed in Fig. 3 are defined below. For simplicity, we consider a situation with one rehabilitation clinic.

INI Initializes the model randomly based on the inputs and possible labels

ENC Cryptographic encryption function with homomorphic properties.

DEC Cryptographic decryption function with homomorphic properties.

EFW Encrypted forward propagation, which forward propagates the input over an encrypted model, and outputs the encrypted results.

EBP Encrypted backpropagation, which backpropagates the encrypted model and outputs an updated, encrypted model.

CMB function that takes two models and combines them into one model.

SPT function that splits a model into partial models.

3 Experiments

3.1 Datasets

St. Antonius Dataset: CVA Outcome Prediction. The goal of the use case is to predict functional status in the form of a dichotomized modified Rankin scale (mRS) score of the CVA patients after 3 months at discharge, based on data collected retrospectively in the hospital and subsequent rehabilitation clinics. The hospital data was based on the Dutch Acute Stroke Audit (DASA) [KWvdBV+18], a stroke national registry. An overview of the data can be found in Table 1. Data from patients that were admitted to the CVA unit of the St. Antonius Hospital in the Netherlands between October 2018 and December 2019 were included in this study *(n = 1014)*. Three (geriatric) rehabilitation clinics, that patients were discharged to, provided data about patients' rehabilitation trajectory. Three months after the CVA, all patients or their caregivers were

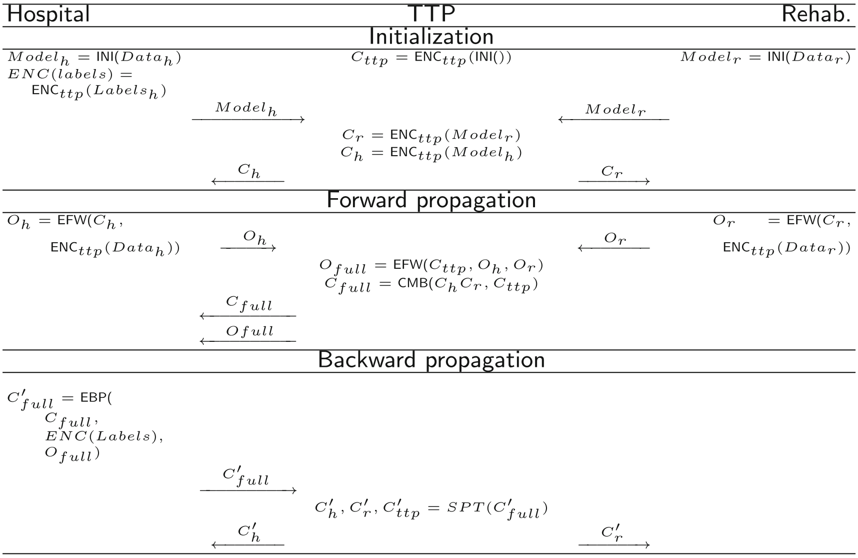

Hospital
TTP
Rehab.
Initialization
$Model_h$ = INI($Data_h$)
$ENC(labels)$ = ENC$_{ttp}$($Labels_h$)
C_{ttp} = ENC$_{ttp}$(INI())
$Model_r$ = INI($Data_r$)
$Model_h$
$Model_r$
C_r = ENC$_{ttp}$($Model_r$)
C_h = ENC$_{ttp}$($Model_h$)
C_h
C_r
Forward propagation
O_h = EFW(C_h, ENC$_{ttp}$($Data_h$))
O_h
O_r
O_r = EFW(C_r, ENC$_{ttp}$($Data_r$))
O_{full} = EFW(C_{ttp}, O_h, O_r)
C_{full} = CMB($C_h C_r$, C_{ttp})
C_{full}
$Ofull$
Backward propagation
C'_{full} = EBP(C_{full}, $ENC(Labels)$, O_{full})
C'_{full}
$C'_h, C'_r, C'_{ttp} = SPT(C'_{full})$
C'_h
C'_r

Fig. 3. Communication and secure processing protocol. O refers to the encrypted outputs of the models, C refers to the encrypted models. Forward and backward propagation are repeated until convergence.

contacted by phone for a follow-up to determine functional status. Patients who died during hospital stay, could not be reached for, or opted out of the three-month follow-up were excluded from the study *(n = 189)*, leaving 825 patients. The mRS score represents functional outcome on a scale from 0 (no residual symptoms) to 5 (severe disability), with 6 signifying the patient has passed away. The scores were dichotomized to favourable outcomes (0–2) and unfavourable outcomes (3–6). Patient data was matched between the hospital and rehabilitation based on personal identifiers, or if not available, on other known characteristics such as gender and age. For patients who did not receive inpatient rehabilitation, days in rehabilitation and rehabilitation time were set to 0. Missing hospital data was imputed using simple imputation, and missing rehabilitation data was imputed with simple imputation with grouped averages based on the type of dismissal.

Additional Public Datasets. To provide a more complete analysis of our SVFL set-up, we also applied our protocol to the following two medical benchmarks public datasets related to CVA, as summarized in Table 2. *CVA prediction* [Kag23] predicts the likelihood that a person will get a CVA based on medical and lifestyle data. *Public CVA rehabilitation* [OOK16] follows patients' rehabilitation after CVA. The dataset includes demographics, medical history, questionnaires, and medical examinations at multiple time points in a year-long observational study. These datasets are suitable as vertically partitioned data and more detailed information can be found in [AKBVH22].

3.2 Experimental Setup

We compared our SVFL setup to several baseline prediction models, creating the following 5 scenarios:

1. *Ideal scenario*: Fully centralized model trained and evaluated on a central repository of both hospital and rehabilitation data
2. *Hospital scenario*: Centralized model trained and evaluated on only hospital data set
3. *Rehabilitation scenario*: Centralized model trained and evaluated on only rehabilitation data

Table 1. Patient characteristics. Abbreviations: FS = functional status (Barthel score), MSR = Medical Specialist Rehabilitation, NH = nursing home, NIHSS = National Institutes of Health Stroke Scale %nm = percentage of non-missing data, O/U = other/unknown, *of patients who had rehabilitation ** FS in rehabilitation uses a different metric (Barthel) than mRS

Hospital data				Rehabilitation data			
		Avg(std)	%nm			Avg(std)	%nm*
Age		72.4(14.1)	99	Time in rehab. (days)		56.4(38.0)	100
NIHSS		6.0(6.1)	94	FS admission**		12.2 (5.6)	93
Time to hosp. (hrs)		17.1(174)	100	FS discharge**		17.5 (4.6)	81
Time in hosp. (days)		7.1(8.2)	100	Active treatment (min)		198 (22.8)	100
Time to IVT (min)		45.0(26.6)	20				
Time to IAT (min)		81.6(73.9)	15				
		n(%)				n(%)	
Gender (Male)		57	99	Type of rehabilitation	None	75	
Atrial Fibrillation		19	63		MSR	7.5	
Referred		17	99		Geriatric	17.5	
Discharged to	Home	49		Discharged to	Home	76	
	NH	1.6			NH	12	
	MSR	7.6			Rehab.	2.5	
	Hospital	8.6			Hospital	2.5	
	Geriatric	18			O/U	7	
	O/U	7.9					
Intra-arterial therapy (IAT)		16	100				
Intravenous thrombolysis (IVT)		21	100				
CVA upon wake-up		22	99				

Table 2. Additional datasets. FeatureSplit refers to the number of features in the first partial dataset.

Dataset	# Samples	# Features	FeatureSplit	% pos. lab.
[OOK16]	1219	200	141	26
[Kag23]	5110	12	5	5

4. *Unsecured scenario (VFL)* Non-encrypted VFL model trained and evaluated on hospital and rehabilitation data, adapted from [AKBVH22]
5. *SVFL*: Our proposed scenario, based on the protocol in Sect. 2.

The SVFL protocol and the four baselines were implemented for our St. Antonius dataset and the two public datasets. Models were multilayer perceptrons, hyperparameter tuning was performed with Optuna [ASY+19] for all three centralized models. Distributed models were based on the finetuned centralized model for the full dataset. Only scenarios of vertically partitioned data among 2 nodes (hospital and rehabilitation center) were considered. For the split of the model among the local nodes, the division in terms of nodes per layer was proportional to the feature division among the nodes. The dataset samples were divided 8:1:1, for the training, validation, and test sets respectively. The experiments were evaluated for predictive performance in AUC and speed in seconds. Experiments were performed in Python 3.7, with the unencrypted models developed with PyTorch 1.8.1 and the encrypted models with CrypTen 0.4.0 [KVH+21], on i5 12th Generation Intel CPU with 8 GB RAM, running the Windows Subsystem for Linux.

4 Evaluations

4.1 Predictive Performance

We evaluated the 5 different scenarios with the aforementioned three datasets, results can be seen in Table 3. While the predictive performance differs between the datasets, we see a similar pattern over the 5 scenarios for each dataset: The centralized model on the full dataset gives the best performance, and the models only on the partial datasets lead to the worst performances. The VFL scenario provides an increase in predictive performance compared to the best-performing partial model, with an AUC of respectively 0.81, 0.94, and 0.81 for [OOK16,Kag23], and the St. Antonius dataset. There is a decrease in predictive performance compared to the 'ideal' scenario, the fully centralized dataset, of respectively 0.01, 0.02, and 0.03. As expected, we saw no difference between the secure and non-secure versions of VFL in terms of AUC.

Table 3. Area under the ROC curve for the 5 different scenarios on the 3 datasets

Dataset	Centralization	Hospital	Rehab.	VFL	Secure VFL
[OOK16]	0.82	0.79	0.74	0.81	0.81
[Kag23]	0.96	0.92	0.90	0.94	0.94
St.Antonius	0.84	0.77	0.76	0.81	0.81

4.2 Feature Importance

To provide insight into the cause of the small differences in model performances between the centralized and federated scenarios, we calculated feature importances. To this purpose, we used SHAP, a method that provides local explainability based on a game theory approach [LL17]. We give an overview of the feature importances for the fully centralized and the VFL scenario for the St. Antonius dataset. The beeswarm plot of the ten most important features in both scenarios can be seen in Fig. 4. We can see that there are small differences between the feature importances, with intra-arterial therapy (IAT) being part of the top 5 predictors for the centralized scenario, whereas in the VFL scenario, this is replaced by whether the patient had a CVA at wake-up.

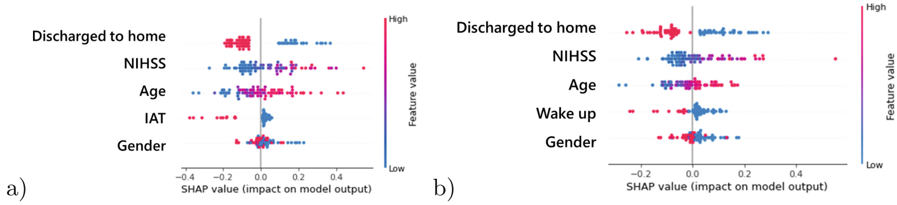


Fig. 4. shows the 5 top features of (a) centralized scenario and (b) vertically federated scenario of the St. Antonius dataset

4.3 Efficiency

We compare unencrypted VFL (scenario 4) with secure VFL (scenario 5) to see the added computational complexity of the secure scenario. For [Kag23, OOK16] and our dataset, we see that the unencrypted scenario respectively took an average of 0.061, 0.045, and 0.031 s per epoch, whereas the SVFL scenario took 88.7, 23.0, and 16.7 s. This implies that the encryption increases the computational time by orders of 2,5–3 of magnitude, which is comparable to the results found in [KVH+21]. We saw the increase was the largest for [Kag23], which contained the largest sample size.

5 Discussion

Overall, the experiments show that the VFL scenarios have added benefit for all tested datasets compared to their partial counterparts. For some datasets, there is a slight decrease in predictive performance compared to the fully centralized scenario. Moreover, we show a more secure version of VFL, at the cost of efficiency. These two points highlight that SVFL is especially a good alternative in situations where centralization of data is not a possibility. This is also true for the St. Antonius dataset we have collected. There is an added benefit of combining the partial models for CVA rehabilitation prediction, where the VFL, both secure and non-secure, outperformed the centralized scenarios of only one partial dataset. We noticed a difference in the feature importances between the centralized and the vertically partitioned model. This could explain the small drop in performance, where the VFL network might struggle to extract the predictive value of certain features and focused on learning from other features instead. Next to better predictions, this also offers the opportunity to provide predictions at different points in time, by improving both the predictions at the rehabilitation clinic and the hospital. We previously found that there is a need for patients to have access to such predictions, also at different points in their rehabilitation process [AvHH+24]. It is still essential to have agreements and proper infrastructure between the different care institutions, but the avoidability of data sharing that SVFL offers could make this easier to facilitate. Moreover, SVFL could also be applicable in other domains than healthcare where sensitive personal data is vertically distributed, like the financial sector.

There are some limitations to this study that have to be considered, both with our collected dataset, as well as with the developed SVFL framework. We were only able to collect a relatively small dataset for deep learning purposes and data quality was affected by data collection being retrospective and some rehabilitation clinics having privacy restrictions on the sharing of data. While both these issues highlight the necessity of an SVFL framework, having a more complete and high-quality dataset could have significantly impacted the predictive performances among the different scenarios. Other limitations of the SVFL framework are concerning the secure Crypten-based aspect of the framework. The security is limited in our setup, as it is honest-but-curious which does not protect against a malicious adversary [KVH+21]. Moreover, due to the computationally expensiveness of the framework, an extensive vertically federated hyperparameter search is currently not possible in a realistic timeframe. As such, a more efficient SVFL could offer opportunities to further increase predictive performance. Future research is needed in several other areas. For example, full decentralization, without a third party, would be preferable to a TTP by limiting the risk for the involved parties. Moreover, previous studies have shown that including imaging, such as CT-perfusion, would lead to an increase in predictive performance for CVA rehabilitation [BAvH24]. Creating an SVFL that allows for imaging data, would require a more efficient protocol combined with a split network that could handle different data modalities.

References

[AKBVH22] Allaart, C.G., Keyser, B., Bal, H., Van Halteren, A.: Vertical split learning-an exploration of predictive performance in medical and other use cases. In: 2022 International Joint Conference on Neural Networks (IJCNN), pp. 1–8. IEEE (2022)

[ASY+19] Akiba, T., Sano, S., Yanase, T., Ohta, T., Koyama, M.: Optuna: a next-generation hyperparameter optimization framework. In: Proceedings of the 25th ACM SIGKDD International Conference, pp. 2623–2631 (2019)

[AvHH+24] Allaart, C.G., et al.: The meaning of a cva outcome prediction model for patients, family and health care providers: a qualitative evaluation. JMIR Preprints. 22/01/2024:56521 (2024)

[BAvH24] Borsos, B., Allaart, C.G., van Halteren, A.: Predicting stroke outcome: a case for multimodal deep learning methods with tabular and ct perfusion data. Artif. Intell. Med. **147**, 102719 (2024)

[BIJC+20] Ben-Israel, D., et al.: The impact of machine learning on patient care: a systematic review. Artif. Intell. Med. **103**, 101785 (2020)

[CSM+20] Ceballos, I., et al.: Splitnn-driven vertical partitioning. arXiv preprint arXiv:2008.04137 (2020)

[fDCP23] Centers for Disease Control and Prevention. Treat and recover from stroke (2023)

[FXC+22] Fu, F., Xue, H., Cheng, Y., Tao, Y., Cui, B.: Blindfl: vertical federated machine learning without peeking into your data. In: Proceedings of the 2022 International Conference on Management of Data, pp. 1316–1330 (2022)

[HLJ+21] Hsu, K.-C., et al.: Comparison of outcome prediction models post-stroke for a population-based registry with clinical variables collected at admission vs. discharge. Vessel Plus **5** (2021)

[JCH+21] Jin, X., Chen, P.-Y., Hsu, C.-Y., Chia-Mu, Yu., Chen, T.: Cafe: catastrophic data leakage in vertical federated learning. Adv. Neural. Inf. Process. Syst. **34**, 994–1006 (2021)

[Kag23] Kaggle. Stroke prediction set (2023)

[KVH+21] Knott, B., Venkataraman, S., Hannun, A., Sengupta, S., Ibrahim, M., van der Maaten, L.: Crypten: secure multi-party computation meets machine learning. Adv. Neural. Inf. Process. Syst. **34**, 4961–4973 (2021)

[KWvdBV+18] Kuhrij, L., Wouters, M., van den Berg-Vos, R., de Leeuw, F.-E., Nederkoorn, P.: Dasa: benchmarking acute stroke care in the Netherlands. Europ. Stroke J. **3**(4), 361–368 (2018)

[LL17] Lundberg, S., Lee, S.-I.: A unified approach to interpreting model predictions (2017)

[LWXO21] Luo, X., Wu, Y., Xiao, X., Ooi, B.C.: Feature inference attack on model predictions in vertical federated learning. In: 2021 IEEE 37th International Conference on Data Engineering (ICDE), pp. 181–192. IEEE (2021)

[NHDC23] Naseri, M., Han, Y., De Cristofaro, E.: Badvfl: backdoor attacks in vertical federated learning. arXiv preprint arXiv:2304.08847 (2023)

[OOK16] Ostir, G., Ottenbacher, K., Kuo, Y.: Stroke recovery in underserved populations 2005-2006. Inter-university consortium for political and social research. Ann Arbor, MI (2016)

[PT06] Porter, M.E., Teisberg, E.O.: Redefining health care: creating value-based competition on results. Harvard Business Press (2006)

[YHL+22] Yang, W., He, Z., Li, Y., Zhang, H., Guan, Z.: A hybrid secure two-party protocol for vertical federated learning. In: Wang, G., Choo, KK.R., Wu, J., Damiani, E. (eds.) UbiSec 2022. LNCS, pp. 38–51. Springer, Cham (2022). https://doi.org/10.1007/978-981-99-0272-9_3

Smoking Status Classification: A Comparative Analysis of Machine Learning Techniques with Clinical Real World Data

Amila Kugic[1], Akhila Abdulnazar[1,2], Anto Knezovic[1], Stefan Schulz[1], and Markus Kreuzthaler[1](✉)

[1] Institute for Medical Informatics, Statistics and Documentation,
Medical University of Graz, Graz, Austria
markus.kreuzthaler@medunigraz.at

[2] CBmed GmbH Center for Biomarker Research in Medicine, Graz, Austria

Abstract. Electronic health records often lack consistent and organized documentation regarding lifestyle-related risk factors. This study addresses this by presenting methodologies aimed at standardizing the recording of patients' smoking status. Different types of machine learning methods are applied to an anonymized set of German-language clinical narratives in order to categorize smoking status as a multi-class classification task utilizing SNOMED CT as a terminology standard. Our findings demonstrate the effectiveness of downstreaming medBERT.de, an openly available medical language model in German, achieving the best performance with an F1-measure of [0.969–0.976] 95% CI, in comparison to CNN, LSTM and an SVM baseline.

Keywords: Natural Language Processing · Electronic Health Records · Machine Learning

1 Introduction

Smoking tobacco is still one of the leading causes of chronic obstructive pulmonary disease (COPD) [12], and was placed third in the top causes of death by the World Health Organization in 2019 [24]. Since then, the introduction and uptake of e-cigarettes have caused an increase in tobacco use, especially in a younger generation, switching from e-cigarettes to conventional ones [22]. The act of smoking and the smoke itself have secondary factors besides affecting airways, lungs, oral and dental health of smokers, and introduce changes to the human body in various forms, e.g., metabolic changes and mental health related challenges. In clinical information systems, details about a patient's smoking status are frequently confined to the narrative content of electronic health records (EHRs). These texts are often riddled with jargon and abbreviations, posing challenges for automated extraction of smoking status in order to obtain structured data. In 2022, an in-depth analysis by Kukhareva et al. [11] of relevant

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 182–191, 2024.
https://doi.org/10.1007/978-3-031-66538-7_19

smoking information from EHR data revealed that in 80% of patient records, various types of inaccuracies and data inconsistencies resulted in an underestimation for lung cancer screening. Examples of inaccuracies were missing data, outdated information or missing metadata, such as quit year or pack years estimations.

Particularly with smoking status as part of social determinants of health [5], studies have shown that the application of machine learning (ML) methods can positively impact prediction and classification tasks [20,26]. In this work, we undertook the task of distinguishing between different kinds of patients' smoking habits, moving beyond a simple binary classification into smokers and non-smokers. Addressing the needs of physicians, a scale was created to establish more nuance in the differentiation of smoking habits of patients, based on the clinical narrative data and biomedical literature in this field. In conjunction, an annotated free-text German dataset was coupled with classical ML and deep learning (DL) methods for automated smoking status classification and evaluation.

2 Related Work

Yang et al. [25] extracted 200 clinical narratives from the University of Florida Health Integrated Data repository to build a natural language processing (NLP) tool for smoking status classification. The aim was to extract smoking-related quantitative data to better gauge the smoking status of a patient, e.g., number of packs per day, active smoking years, quit year, etc. In order to automatically annotate clinical narratives accordingly, a two-layer rule-based approach was employed, i.e., highly specialized rules were created in a two-step process. The system reached a lenient and strict F1-measure of 0.963 and 0.946, respectively, for evaluation on the test set. Ruckdeschel et al. [18] explored the classification of English smoking status mentions into five distinct classes: Active Smoker, Former Smoker, Never-Smoker, History of Smoking, or Unknown. A rule-based extraction with named entity recognition from clinical narratives was combined with a DL sentence classification model to determine the smoking status. To this end, a transformer-based model was used with transfer learning to fine-tune BlueBERT [16], a large language model (LLM), trained on clinical notes from the MIMIC-III [8] database, and the model performed chronological processing of clinical narratives, i.e., the last smoking status mention was favored. During the processing, the extraction of two main criteria was focused on: pack-years smoked, and time from quit date, if applicable. These two factors helped in the classification of patients into Active Smokers, Former Smokers, and Unknown, with sub-categorizations. As the goal was to determine, which patients qualified for low-dose computed tomography, the F1-measure for correctly selecting patients was 0.88. Bae et al. [1] performed a smoking status classification for a bilingual EHR dataset with 4,711 clinical notes in English and Korean. The classification used the classes Current Smoker, Past Smoker, Never-Smoker, and Unknown. A linear support vector machine (SVM) approach trained from unigram and bigram bag-of-words was compared to a keyword expansion and search based mainly on vector similarity, achieving an F1-measure of 0.90. The keyword classification algorithm improved upon the baseline by as much as 1.8%. Caccamisi et al. [3] processed Swedish EHR data to classify sentences into the classes Current Smoker, Ex-Smoker, Non-Smoker, and Unknown. SVM was combined with four macro settings of the tool Weka, i.e., sequential minimal optimization (SMO), k-nearest neighbor, Naive Bayes, and J48 (decision tree). For training, 85,509 rows of information entered into a smoking status text field were processed, while 177,110 rows were utilized for testing. The results showed the best model to be the SVM-SMO model, with an F1-measure of 0.98. Karlsson et al. [9] applied BERT (Bidirectional Encoder Representations from Transformers) [4] and ULMFiT (Universal Language Model Fine-tuning for Text Classification) [7] models for the classification of smoking status for a Finnish cancer cohort of 29,823 patients into the classes Never-Smoker, Former Smoker, Persistent Smoker, and Missing data. The processing consisted of a rule-based extraction of random 5,000 tobacco-related smoking status mentions, which were manually labeled for the training of DL models. If multiple classifications over time were found, a rule-based decision with classification class and metadata was finalized. Both models performed similarly, i.e., the ULMFiT and BERT models reached 0.87 and 0.88 in precision respectively. Rajendran and Topaloglu [17] used English clinical narratives from 781 patients to classify the smoking status according to a binary (Smoker, Never-Smoker) and multi-class classification (Current Smoker, Former Smoker, Never-Smoker) schema. Three classical ML algorithms, Naive Bayes, SVM, and logistic regression, and three DL methods, unidirectional long-short term memory (LSTM), bidirectional LSTM, and convolutional neural network (CNN), were compared. Extensive rule-based preprocessing of clinical narratives to correct language, spelling, as well as removal of punctuation, repeat sentences, and stop words, was executed. For binary classification, the best performance was found in CNN with pre-trained word embeddings, reaching an F1-measure of 0.85. For the multi-class classification schema, CNN still remained the best method in terms of accuracy, with 0.68 in F1-measure, while Naive Bayes performed comparably well, with an F1-measure of 0.69.

3 Data

De-identified German clinical discharge letters from cardiology, dermatology, and oncology were processed with rules to extract smoking related mentions from KAGes, an Austrian Hospital Network. The extraction was supported by an expert in Extract-Transform-Load (ETL)-based processes. Any clinical narratives matching the regular expression `[ikotin|F17|[^bB]rauch|Rauch|Zig]` were included in the initial dataset, with the aim to extract any mention of nicotine, smoke inhalation, smokers, or cigarette consumption. 100 characters to the left and right of the match were extracted, with the signal expression occurring in the middle. With these snippets, a gold standard was created, where each of the snippets was classified by a physician into one of the following six categories:

Past Smoker, Current Smoker, Current Non-Smoker, Never-Smoker, Current or Past Smoker, and Smoking Consumption Unknown [10]. Twenty percent of the dataset was independently re-annotated by a second annotator to calculate the inter-rater agreement [14]. A Cohen's kappa κ of 0.89 indicates a high agreement between the annotators [13]. The categories were created through a bottom-up approach in combining a thorough analysis of the dataset with existent terminologies, i.e., SNOMED CT[1]. Table 1 gives an overview of the classifications and class distributions over the whole dataset.

Table 1. SNOMED CT value set and class distributions.

Class	SCTID	Preferred Term	Counts
0	8517006	Ex-Smoker	2,182
1	77176002	Smoker	4,255
2	8392000	Non-Smoker	27
3	266919005	Never Smoked Tobacco	432
4	410511007; 77176002	Current or Past Smoker	85
5	266927001	Tobacco Smoking Consumption Unknown	261

All classes were assigned SNOMED CT codes. Because Class 4 "Current or Past Smoker" has no SNOMED CT code, it was represented by the combination of the code of "Smoker" with a temporal context value.

4 Methodology

4.1 Overview of Machine Learning Approaches

Four ML approaches were utilized for a comparative analysis of smoking status classification. SVM acts as a baseline, as a classical ML algorithm. SVMs calculate and find the optimal hyperplane that separates classes in a feature space. Additionally, with three different DL models, three aspects of DL for multi-class classification are under investigation with CNN, LSTM and BERT models. CNNs apply layers to extract local features from texts, i.e., local patterns in texts are primarily focused on. LSTM, a type of recurrent neural network, is designed to capture sequential data. Lastly, a transformer-based BERT architecture leverages bidirectional attention mechanisms via a pre-trained language model to understand the context in a sentence.

[1] https://www.snomed.org/five-step-briefing.

4.2 Text Preprocessing and Representation

Line breaks within the contextual information in the texts were removed prior to the extraction of the data. No other pre-processing was performed, particularly no cleaning of misspellings and other non-standard uses of language. For each chosen methodology, different textual representations are needed. For SVM, TF-IDF[2], was applied to have a weighted bag-of-words schema of the line under investigation. For LSTM and CNN, label encoders are used to transform categorical values, i.e., the assigned classes, into numerical values for processing, as well as an embedding representation was leveraged. For BERT, the pre-trained language model "GerMedBERT/medbert-512" [2] trained on German medical texts, clinical narratives and health-related medical information was downstreamed to the problem domain. This particular language model is BERT-based and exploits the multi-layer bidirectional transformer encoder to capture the contextual information present in the dataset.

4.3 Cross-validation, Hyperparameter Tuning and Grid Search

A 10×5 nested cross-validation was implemented to measure how well a trained model type can adapt to a dataset, as well as to give an indication on how generalizable methods truly perform independent of the dataset split. Additionally, cross-validation in conjunction with grid search helped in selecting the best hyperparameters for each method. The selection of the best hyperparameters was based on the metric "accuracy". For SVM, the regularization parameter, type of kernel function and gamma were tuned. For LSTM and CNN, the learning rate and the batch size were varied. For BERT, no hyperparameter tuning was performed, and the base model settings for learning rate and batch size were applied. During training, the goal remained to establish, which of the variations in parameter settings performed the best for each method.

4.4 Model Evaluation

The performance of the best model per random state was measured with the weighted average metrics precision, recall and F1-measure. The weighted average was chosen based on the imbalanced multi-class dataset being processed. Due to the ten chosen random states, the mean and standard error (SE), as well as the confidence intervals (CIs), across all ten random states for each method, are reported on.

4.5 Processing Pipeline

The dataset consists of the context from the clinical narrative and the label that was assigned by the annotators (see Table 1). The separation of the dataset

[2] TF-IDF: term frequency - inverse document frequency.

into training and test was achieved with the train_test_split function from scikit-learn [15], into an 80% train and 20% test set. In conjunction with 10×5 nested cross-validation, 10 different random states[3] were computed. After transforming the textual input data, grid search was utilized for hyperparameter tuning. After training each model and predicting on the test set, the best parameters for each random state were saved. The performance metrics in the results refer to the best performing hyperparameter model between the computed random states.

5 Results

The mean performance metrics per model show that BERT, based on the confidence intervals, significantly outperformed all other methods, with a mean F1-measure of 0.973. CNN followed closely with 0.942, followed by SVM and LSTM, with 0.891 and 0.850, respectively. This trend is also mirrored in mean precision and recall metrics. In Table 2, the final weighted average mean scores with standard errors and confidence intervals are listed.

Table 2. Mean performance metrics for SVM, CNN, LSTM, BERT models on the test data reported with precision, recall and F1-measure.

Classifier	Metrics	Mean ± SE	95% CI
SVM	Precision	0.894 ± 0.002	[0.889 – 0.900]
	Recall	0.894 ± 0.003	[0.888 – 0.900]
	F1-measure	0.891 ± 0.003	[0.885 – 0.897]
CNN	Precision	0.951 ± 0.003	[0.944 – 0.957]
	Recall	0.940 ± 0.002	[0.934 – 0.945]
	F1-measure	0.942 ± 0.002	[0.937 – 0.948]
LSTM	Precision	0.866 ± 0.004	[0.856 – 0.875]
	Recall	0.845 ± 0.005	[0.834 – 0.856]
	F1-measure	0.850 ± 0.006	[0.838 – 0.862]
BERT	Precision	0.973 ± 0.002	[0.970 – 0.976]
	Recall	0.972 ± 0.002	[0.970 – 0.975]
	F1-measure	0.973 ± 0.002	[0.969 – 0.976]

For hyperparameter tuning, SVM reached optimal performance between all random states with the regularization parameter C at 10, gamma set at scale, and kernel function set to radial basis function. Similarly, for LSTM and CNN, the selection of best parameters resulted in a batch size equal to 64 and 128, respectively, with a learning rate set to 0.01. For BERT, the standard preset values, i.e., batch size at 8 and learning rate set to 0.0004, achieved state-of-the-art results.

[3] [509, 906, 331, 172, 729, 250, 762, 629, 926, 392].

6 Discussion

In 2024, a recent systematic review on smoking status determination by Haque et al. [6], summarized that a majority of articles applied rule-based methods, followed by 44% of articles applying NLP methods, with 29% belonging to SVM models. From the described related work in Sect. 2, we can assume that a pure rule-based approach can reach optimal results above 0.963 in F1-measure. When considering the approaches covered in this paper, the linear SVM approach by Bae et al. [1] reached a similar performance of 0.900 in F1-measure as the SVM method employed by us, with a mean F1-measure of 0.891. Similarly, Caccamisi et al. [3] have shown that an optimized SVM model would even be able to improve upon the baseline. Rajendran and Topaloglu [17] compared classical and DL algorithms for a multi-class classification schema. CNN outperformed all other methods, including SVM, Naive Bayes, logistic regression, and LSTM with an F1-measure of 0.690. Comparably, the same methodology for our dataset reached a high mean F1-measure of 0.942, and similarly outperformed the LSTM method with 0.850. Without the application of the BERT methodology, CNN would have resulted in the best performance, as reported on by Rajendran and Topaloglu [17]. However, the transformer-based architecture for our German-language dataset was applied, and it outperformed all other methods. Ruckdeschel et al. [18] stated that a fine-tuned BERT model, trained on clinical notes in English, resulted in an F1-measure of 0.880, while Karlsson et al. [9] similarly applied a language-specific BERT model with a performance of 0.880 in precision. Especially regarding the characteristics of German-language clinical narratives, which are complex and filled with short forms and jargon expressions [19], classifications of smoking status or other social determinants of health can be ambiguous. Most information in clinical texts rely on contextual information to be understood. The applied pre-trained language model and BERT method seemed to increase contextual understanding, and resulted in a mean F1-measure of 0.973.

6.1 Error Analysis

For better understanding of the performance results per method, a summary error analysis of the best performing model per method was done. On class level, SVM and BERT methods accomplished a robust and high performance across all classes. LSTM and CNN both seemed to have one class in the multi-class classification schema, which was either missed completely or performed very poorly. LSTM completely missed the “Current or Past Smoker” class, which mainly consisted of two contextual descriptions: (i) prescription of nicotine products, e.g., “Nikotinell 14 mg in 24 Std” (nicotine patch with 14mg in strength per 24 h), or (ii) patient reported nicotine abstinence with quit year estimation, e.g., “seit 5 Jahren Nikotin-Karenz” (past 5 years nicotine abstinence). Furthermore, LSTM also has shown slight decreases with classes that are less frequently represented in the dataset, i.e., dataset imbalance seems to have a larger effect with LSTM compared to other models. As for CNN, the class "Current Non-Smoker" performed very poorly, with a weighted F1-measure of 0.213. Most entries in the test set were classified incorrectly, and from the false positives and negatives, the compound noun word "Nichtraucher" ("non-smoker"), was not contextually understood by the CNN model.

6.2 System Limitation

The models were trained with narrative content from Austrian EHRs, which is why the application is limited to the German language. Furthermore, deviations from medical practice in the description of nicotine status cannot be recognized by the models, as these do not occur in the data set. As the data was only collected from three clinical departments, a selection bias can be assumed. Depending on the medical specialty, documentation of smoking status are handled differently, or even collected in the first place. Further data from a pulmonology and an angiology department would have been interesting, because of a much higher smoking prevalence of their patients, who typically suffer from smoking-related diseases.

7 Conclusion and Outlook

In this paper, a comparative analysis of ML techniques was performed to automatically classify smoking status mentions in clinical narratives with SVM, LSTM, CNN, and BERT methods. Cross-validation, hyperparameter tuning and grid search were applied for optimal customization and robust evaluation results. BERT outperformed all other methodologies and reached state-of-the-art results, with CNN, SVM and LSTM following, in that order.

Future work will focus on enhanced robustness of the models with data from other clinical departments, but also from other countries and languages. Thus, varied practices of clinical documentation would also be reflected in the training dataset. Besides a first positive evaluation on the performance of the classification task, a technical integration of an UIMA-based [21] natural language processing component is planned as a next step.

In accordance with the FAIR criteria (Findable, Accessible, Interoperable, Reusable) [23], a consideration of smoking status in the context of HL7-FHIR, in combination with SNOMED CT would be of high interest for international standardization of lifestyle data, particularly those related to global health, with tobacco smoking being a leading cause of preventable diseases, such as lung cancer, heart disease and stroke.

References

1. Bae, Y.S., et al.: Keyword extraction algorithm for classifying smoking status from unstructured bilingual electronic health records based on natural language processing. Appl. Sci. **11**(19), 8812 (2021). https://doi.org/10.3390/app11198812, https://www.mdpi.com/2076-3417/11/19/8812

2. Bressem, K.K., et al.: medbert.de: a comprehensive German bert model for the medical domain. Expert Syst. Appl. **237**, 121598 (2024). https://doi.org/10.1016/j.eswa.2023.121598, https://www.sciencedirect.com/science/article/pii/S0957417423021000
3. Caccamisi, A., Jørgensen, L., Dalianis, H., Rosenlund, M.: Natural language processing and machine learning to enable automatic extraction and classification of patients' smoking status from electronic medical records. Upsala J. Med. Sci. **125**(4), 316–324 (2020)
4. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, May 2019. https://doi.org/10.48550/arXiv.1810.04805, http://arxiv.org/abs/1810.04805, arXiv:1810.04805 [cs]
5. Hacker, K., Houry, D.: Social needs and social determinants: the role of the centers for disease control and prevention and public health. Public Health Rep. **137**(6), 1049–1052 (2022). https://doi.org/10.1177/00333549221120244, publisher: SAGE Publications Inc
6. Haque, M.A., Gedara, M.L.B., Nickel, N., Turgeon, M., Lix, L.M.: The validity of electronic health data for measuring smoking status: a systematic review and meta-analysis. BMC Med. Inform. Decision Making **24**(1), 33 (2024). https://doi.org/10.1186/s12911-024-02416-3. https://bmcmedinformdecismak.biomedcentral.com/articles/10.1186/s12911-024-02416-3
7. Howard, J., Ruder, S.: Universal language model fine-tuning for text classification. In: Gurevych, I., Miyao, Y. (eds.) Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 328–339. Melbourne, Australia (2018). https://doi.org/10.18653/v1/P18-1031, https://aclanthology.org/P18-1031
8. Johnson, A.E., et al.: MIMIC-III, a freely accessible critical care database. Sci. Data **3**(1), 160035 (2016). https://doi.org/10.1038/sdata.2016.35. https://www.nature.com/articles/sdata201635
9. Karlsson, A., et al.: Impact of deep learning-determined smoking status on mortality of cancer patients: never too late to quit. ESMO Open **6**(3), 100175 (2021). https://doi.org/10.1016/j.esmoop.2021.100175. https://linkinghub.elsevier.com/retrieve/pii/S2059702921001356
10. Knezovic, A.: Extraction and standardization of smoking status from free-text clinical routine documentation using machine learning methods. Master's thesis, Medical University of Graz (2023)
11. Kukhareva, P.V., et al.: Inaccuracies in electronic health records smoking data and a potential approach to address resulting underestimation in determining lung cancer screening eligibility. J. Am. Med. Inform. Assoc. **29**(5), 779–788 (2022). https://doi.org/10.1093/jamia/ocac020, https://academic.oup.com/jamia/article/29/5/779/6529026
12. Lu, W., et al.: Tobacco and chronic obstructive pulmonary disease (COPD). World Health Organization, November 2023. https://www.who.int/publications-detail-redirect/9789240084452
13. McHugh, M.L.: Interrater reliability: the Kappa statistic. Biochemia medica **22**(3), 276–282 (2012)
14. O'Connor, C., Joffe, H.: Intercoder reliability in qualitative research: debates and practical guidelines. Int J Qual Methods **19**, 1609406919899220 (2020)
15. Pedregosa, F., et al.: Scikit-learn: machine learning in Python. J. Mach. Learn. Res. **12**, 2825–2830 (2011)

16. Peng, Y., Yan, S., Lu, Z.: Transfer learning in biomedical natural language processing: an evaluation of bert and elmo on ten benchmarking datasets. In: Proceedings of the 2019 Workshop on Biomedical Natural Language Processing (BioNLP 2019), pp. 58–65 (2019)
17. Rajendran, S., Topaloglu, U.: Extracting smoking status from electronic health records using NLP and deep learning. AMIA Summits Transl. Sci. Proc. **2020**, 507–516 (2020). https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7233082/
18. Ruckdeschel, J.C., Riley, M., Parsatharathy, S., Chamarthi, R., Rajagopal, C., Hsu, H.S., Mangold, D., Driscoll, C.: Unstructured Data Are Superior to Structured Data for Eliciting Quantitative Smoking History From the Electronic Health Record. JCO clinical cancer informatics **7**, e2200155 (Feb 2023). https://doi.org/10.1200/CCI.22.00155
19. Schwarz, C.M., et al.: Structure, content, unsafe abbreviations, and completeness of discharge summaries: a retrospective analysis in a University Hospital in Austria. J. Eval. Clin. Practice **27**(6), 1243–1251 (2021). https://doi.org/10.1111/jep.13533, https://onlinelibrary.wiley.com/doi/10.1111/jep.13533
20. Stabellini, N., et al.: Social determinants of health data improve the prediction of cardiac outcomes in females with breast cancer. Cancers **15**(18), 4630 (2023). https://doi.org/10.3390/cancers15184630, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10526347/
21. The Apache Software Foundation: UIMA. Unstructured Information Management Architecture (2010). https://uima.apache.org/
22. WHO: The European Health Report 2021. Taking stock of the health-related Sustainable Development Goals in the COVID-19 era with a focus on leaving no one behind. World Health Organization, March 2022. https://www.who.int/europe/publications/i/item/9789289057547
23. Wilkinson, M.D., et al.: The FAIR guiding principles for scientific data management and stewardship. Sci. Data **3**(1) (2016). https://doi.org/10.1038/sdata.2016.18, https://doi.org/10.1038/sdata.2016.18
24. World Health Organization (ed.): European health report 2018: more than numbers - evidence for all. Regional Office for Europe, Copenhague (2018). https://iris.who.int/handle/10665/279904
25. Yang, X., et al.: A natural language processing tool to extract quantitative smoking status from clinical narratives. In: 2020 IEEE International Conference on Healthcare Informatics (ICHI), pp. 1–2, November 2020. https://doi.org/10.1109/ICHI48887.2020.9374369. https://ieeexplore.ieee.org/document/9374369, iSSN: 2575-2634
26. Yu, Z., et al.: A study of social and behavioral determinants of health in lung cancer patients using transformers-based natural language processing models. In: AMIA Annual Symposium Proceedings 2021, pp. 1225–1233, February 2022. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8861705/

The Impact of Data Augmentation on Time Series Classification Models: An In-Depth Study with Biomedical Data

Bikram De(✉), Mykhailo Sakevych, and Vangelis Metsis

Texas State University, San Marcos, TX 78666, USA
bikramkumarde@txstate.edu

Abstract. Data augmentation is the practice of applying various transformations to existing data to increase their size and diversity without collecting new data. While augmentation strategies are widely recognized and implemented in image-based deep learning (DL) workflows, the degree to which they are effective in the time series domain is unclear. This paper experimentally evaluates the utility of various common time series augmentation techniques, especially those relevant to the medical sector where data limitations are prevalent. We thoroughly examine popular time series augmentation and synthetic data generation methods to evaluate their effectiveness in downstream classification tasks, encompassing both traditional and DL-based approaches. This research aims to offer insights into the applicability and efficacy of data augmentation strategies in improving model generalization and mitigating data scarcity challenges, with a focus on biomedical time-series data.

Keywords: Time Series · Data Augmentation · Classification

1 Introduction

Data augmentation is a technique for enhancing the size and diversity of training datasets in machine learning. It involves creating modified versions of existing data or synthesizing new data samples based on the statistical properties of the original dataset. This method is instrumental across various data types, including images, audio, video, and text. Our focus, however, narrows down to the application of data augmentation techniques on time series data, a domain that presents unique challenges and opportunities, especially within the medical field.

The infusion of augmented data into the training process of time series models offers significant advantages. It aids in the development of robust, flexible models capable of generalizing effectively to new, unseen data. By introducing a variety of scenarios and patterns through augmentation, models can better learn the complex, non-linear relationships and temporal dependencies that characterize

Project source code: https://github.com/imics-lab/time-series-augmentation.

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 192–203, 2024.
https://doi.org/10.1007/978-3-031-66538-7_20

time series data. This is particularly crucial in mitigating the risks of overfitting and enhancing the performance of models trained on limited, imbalanced, or noisy datasets. Moreover, by limiting augmentation to the training phase, the integrity and authenticity of the data during inference are preserved, ensuring the models' applicability to real-world scenarios remains uncompromised.

However, these advantages come with their own set of limitations. Despite not directly affecting the inference phase, the process of data augmentation must navigate the complexities of medical data's sensitivity, specificity, and multidimensional nature. The creation of augmented data requires careful consideration to avoid introducing biases or artifacts that could mislead the learning process or obscure critical information. Some augmentation techniques that directly apply to other domains, for example, image flipping or rotating, may be ineffective or even detrimental for time series data because they distort the inherent temporal dependencies. Additionally, ethical and privacy concerns are paramount, as the augmentation process involves manipulating sensitive patient data, necessitating stringent adherence to data protection and privacy regulations. In conclusion, while augmenting time series training sets offers a pathway to developing more capable and generalizable models, it necessitates a careful, ethically mindful approach.

In this paper, we present a survey of existing time series augmentation techniques and their effectiveness on different types of time series data. We categorize augmentation techniques into two parts: (1) traditional methods and (2) generative methods (Fig. 1). Traditional methods involve simple signal transformations such as jittering, scaling, magnitude warping, time warping, and window slicing. Augmented copies of the original training samples are added to the training set. Deep learning-based generative methods introduce new data samples by first modeling the statistical properties of the dataset and then generating new data that obey these statistical properties but are not identical to any of the original samples on which the models were trained. The three most popular categories of generative methods at the time of writing are Generative Adversarial Networks (GANs) [7], Variational Autoencoders (VAEs) [1], and Diffusion models [8].

We assess the impact of these augmentation techniques on four distinct time series datasets – pertaining to human activity recognition, sleep studies, heart disorder recognition, and epileptic seizure detection (see Sect. 3.1 for more details) – and use the original and augmented versions of the data for downstream classification tasks with three popular deep learning time series classification architectures, namely LSTM [5], CNN [3], Transformer [14].

Our experimental findings underscore the nuanced effects of different augmentation strategies on model accuracy, influenced by the specific characteristics of the data and the architecture of the classifiers. These insights highlight the absence of a universally optimal augmentation approach, advocating for a tailored selection of techniques based on the specific requirements of each task. A detailed discussion of our empirical observations and their implications for machine learning practice in time series analysis will be elaborated in subsequent sections of this paper.

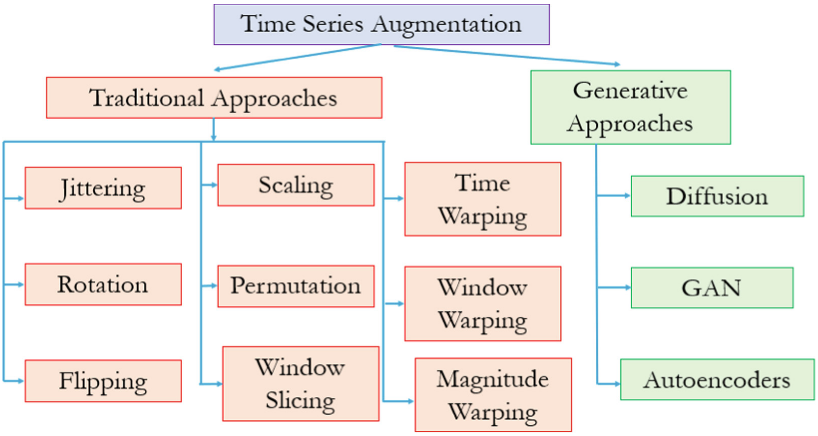


Fig. 1. Overview of various Time series augmentation techniques.

2 Background

The exploration of time series data augmentation techniques has evolved significantly, with various methodologies being developed to enhance the robustness and performance of machine learning models. This body of work encompasses a range of strategies aimed at enriching training datasets, thereby improving model generalization across diverse applications such as classification, forecasting, and anomaly detection. Previous studies, such as those by Wen et al. [12] and Iglesias et al. [2], have provided comprehensive overviews of augmentation methods, discussing their applications, the metrics for evaluation, and the challenges encountered with each technique. Despite these efforts, a gap remains in directly comparing the effects of these augmentation methods across different types of datasets, particularly those related to human activity and medical diagnosis.

In this context, our paper endeavors to bridge this gap by offering a detailed experimental comparison of traditional and deep learning-based generative augmentation techniques. Figure 1 provides an overview of the augmentation techniques compared in this work. We assess their impact on datasets pertinent to human activity recognition and medical diagnosis, employing various model architectures to evaluate the effectiveness of each augmentation method.

2.1 Augmentation Techniques Overview

In this section, we delve into several common augmentation techniques examined in this work and provide a brief formal definition of each.

Rotation: Rotation augmentation involves applying a transformation matrix to the original time series data to generate new samples. This method is mathematically represented as: $X_{\text{rotated}} = R(\theta)X$ where X is the original data, $R(\theta)$ is the rotation matrix defined by the rotation angle θ, and X_{rotated} is the rotated data.

Jittering: Jittering introduces small, random variations to the data, effectively modeled as: $X_{\text{jittered}} = X + \mathcal{N}(0, \sigma^2)$ where X is the original data and $\mathcal{N}(0, \sigma^2)$ represents Gaussian noise with mean 0 and variance σ^2.

Flipping: Flipping reverses the time series data, mathematically described by: $X_{\text{flipped}}[t] = X[N - t]$ where X is the original series, N is the length of the series, and t is the time step.

Scaling: Scaling adjusts the amplitude of the data either by magnifying or by shrinking the data point range of values. $X_{\text{scaled}} = \text{multiplier} \times X_{\text{original}}$ where X is the original data and *multiplier* is a scaling factor which can be either greater or less than 1.

Permutation: Permutation reorders the data points randomly: $X_{\text{permuted}} = X[\pi(i)]$ where X is the original data and π represents a permutation of the indices i.

Window Slicing: Window slicing segments the data into windows, formally: $X_{\text{slice}} = X[t : t + w]$ where X is the original series, t is the starting point, and w is the window size.

Time Warping: Time warping alters the temporal scale: $X_{\text{warped}}(t) = X(\lambda t)$ where X is the original series and λ is the warping factor.

Window Warping: Window warping applies localized transformations: $X_{\text{window warped}} = \text{Transform}(X_{\text{window}})$ where X_{window} is a segment of the original series and Transform denotes the applied warping.

Magnitude Warping: Magnitude warping modifies the amplitude: $X_{\text{magnitude warped}} = X \cdot \lambda$ where X is the original series and λ is the warping factor.

Fourier Transform: Fourier Transform augmentation modifies the frequency components: $\mathcal{F}(X_{\text{augmented}}) = \mathcal{F}(X) + \Delta\mathcal{F}$ where $\mathcal{F}(X)$ is the Fourier transform of the original data, and $\Delta\mathcal{F}$ represents the modifications in the frequency domain.

Generative Adversarial Networks (GANs): GANs generate synthetic data by training a generator G to produce data that a discriminator D cannot distinguish from real data, represented as: $G(z) \approx X$ where z is random noise input and X is the real time series data.

Variational Autoencoders (VAEs): VAEs generate synthetic data by encoding input data X into a latent space z and then decoding it, shown as: $X_{\text{synthetic}} = \text{Decoder}(\text{Encoder}(X))$.

Diffusion Models: Diffusion models represent a class of generative models that gradually transform data from a simple distribution (e.g., Gaussian noise) into complex data distributions by learning to reverse a diffusion process, $X_{t-1} =$

$f(X_t, \theta)$, where X_t represents the data at step t, X_{t-1} is the data at the previous step, and f is a learned function parameterized by θ. In the context of time series data augmentation, diffusion models can be employed to generate synthetic time series data that captures the intricate temporal dynamics and distributions of the original dataset.

3 Methodology

Our methodology encompasses a rigorous approach to evaluating the efficacy of various data augmentation techniques applied to time series datasets, ensuring the integrity of our experimental setup and the reliability of our results. To address the challenges inherent in time series data analysis, particularly the risk of data leakage, we meticulously partition the datasets based on subjects. This strategy guarantees that each subject is exclusively included in either the training or testing set, thereby preserving the independence of our test data and ensuring it remains unseen during training.

In our analysis, we explore both traditional and deep learning-based augmentation methods. Traditional techniques such as jittering, scaling, magnitude warping, time warping, window warping, and window slicing are systematically evaluated. Each technique is parameterized to quantify the extent of augmentation, with experiments conducted across a spectrum of parameters using 10-fold cross-validation. This process allows us to identify the parameter setting that maximizes mean accuracy for each augmentation method. Using the selected parameter, we augment the training data doubling the size of each class. Thus the entire training data is doubled and the class ratios remain unaltered. Using the best augmentation hyperparameter, we again shuffle the data 10 times and calculate the accuracy for each shuffle. Using the 10 accuracies, we calculate the technique's average accuracy and 95% confidence intervals.

For the task of classification, we leverage three distinct classifier models: LSTM, CNN, and Transformer-based models, utilizing the TSAI library [11] for state-of-the-art implementations. Specifically, we employ the LSTM-FCN architecture [5] for the LSTM model, the Inception Time model [3] for the CNN, and the TST architecture [14] for the Transformer model, as implemented in the TSAI library [11].

Data preprocessing forms the initial phase of our methodology, where data from four distinct datasets are prepared for analysis. This involves loading data from various channels, processing it through data loaders, and splitting it into training, testing, and validation sets based on subjects. This subject-based splitting is critical for avoiding data leakage in time series analysis.

Our augmentation pipeline is depicted in Fig. 2 for traditional methods. In contrast, the pipeline for deep learning-based methods differs in the final step by eliminating the need for multiple shuffling iterations, instead requiring only a single iteration of data generation. We apply six traditional augmentation methods, sourcing implementations [4]. For deep learning-based augmentation, we investigate three techniques: a Transformer-based GAN [7], a Variational Autoencoder [1], and a Diffusion model [8], each implemented from recent literature.

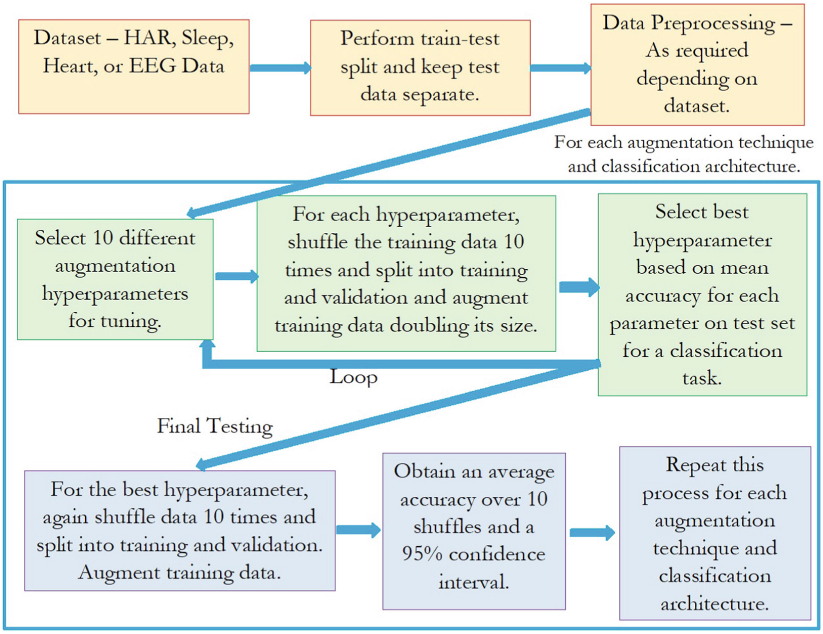

Dataset – HAR, Sleep, Heart, or EEG Data
Perform train-test split and keep test data separate.
Data Preprocessing – As required depending on dataset.
For each augmentation technique and classification architecture.
Select 10 different augmentation hyperparameters for tuning.
For each hyperparameter, shuffle the training data 10 times and split into training and validation and augment training data doubling its size.
Select best hyperparameter based on mean accuracy for each parameter on test set for a classification task.
Loop
Final Testing
For the best hyperparameter, again shuffle data 10 times and split into training and validation. Augment training data.
Obtain an average accuracy over 10 shuffles and a 95% confidence interval.
Repeat this process for each augmentation technique and classification architecture.

Fig. 2. Overview of augmentation and testing pipeline for traditional time series augmentation techniques.

Through this comprehensive methodology, we aim to provide a detailed comparison of the impact of various augmentation techniques on time series datasets, focusing on human activity recognition and medical diagnosis. Our approach ensures a robust evaluation framework, leveraging advanced classification models to assess the effectiveness of each augmentation technique in enhancing dataset quality and model performance.

3.1 Dataset Description

The four datasets described below were selected as representative of typical biomedical applications using machine learning models.

Human Activity Recognition: The UniMiB SHAR [9], is a dataset of acceleration samples acquired with an Android smartphone designed for human activity recognition and fall detection. The dataset includes 11,771 samples of both human activities and falls performed by 30 subjects of ages ranging from 18 to 60 years. The dataset contains 9 types of daily living activities. The 9 types of daily living activities include: Standing Up From Sitting, Standing Up From Laying, Walking, Running, Going Upstairs, Jumping, Going Downstairs, Lying Down From Sitting, Sitting Down.

Sleep Event Detection: The Polysomnography (PSG) dataset [6] used in this work contains data recorded on 212 individuals in a hospital setting for sleep apnea syndrome (SAS) diagnosis. Five categories of abnormal events were annotated by a medical team ("respiratory", "neurological", "limb activity related", "nasal", and "cardiac"). In this study, we detect only the respiratory events, thus forming a binary classification task. We use only the 12 signal channels that are most relevant to the respiratory events. Nine of the channels (3 x EEG, 2 x EMG, 2 x EOG, 2 leg sensors, and ECG) were downsampled from 200 Hz to 100 Hz to match the remaining three sensors used (flow thermistor plus thoracic and abdominal respiratory belts).

Heart Disorder Detection: The MIT-BIH disorder dataset [10] contains 48 snippets of ambulatory ECG recordings spanning half an hour each from 47 subjects across five heart conditions. The samples, originally recorded at 125 Hz, have been adjusted to 187 in length for U-Net compatibility. The training set has 87554 samples, with the majority class having 72471 samples and the smallest class having 641. The test set includes 21892 samples, ranging from 162 to 18118 samples per class. The majority class of both the training and testing set was reduced to 10% to prevent class imbalance.

Epileptic Seizure Recognition: This dataset [13] consists of 5 different folders, each with 100 files, with each file representing a single subject/person. Each file is a recording of brain activity for 23.6 s. The corresponding time-series is sampled into 4097 data points. Each data point is the value of the EEG recording at a different point in time. So we have a total of 500 individuals, with each having 4097 data points for 23.5 s. The five different folders represent five different situations in which the EEG signal is recorded from the brain. The folders include eyes open, and eyes closed, recordings from healthy brain areas with a tumor in the brain, recordings from the part of the brain with the tumor, and the last folder, recordings of seizure activity. A binary classification is performed with this data for recording of seizures against others.

4 Results

In Fig. 3, we show the downstream classification of each combination of augmentation technique and classification architecture on the four datasets used in this study. Each of the four plots shown in the figure corresponds to a different dataset, as indicated by the label on the top of the plot. Each plot is a bar chart, with the bars separated into three groups corresponding to the three deep-learning classification architectures. Each bar within a group corresponds to a different augmentation technique, showing the mean accuracy accomplished when applying that augmentation technique to data. Along with the accuracy, a 95% confidence interval range is shown at the top of each bar. The order is maintained across groups and plots for comparison consistency.

Due to space limitations, only the plots are shown here. For the detailed numeric results in tabular format and specific parameter values used by each augmentation technique, the reader should refer to the Appendix of this paper.

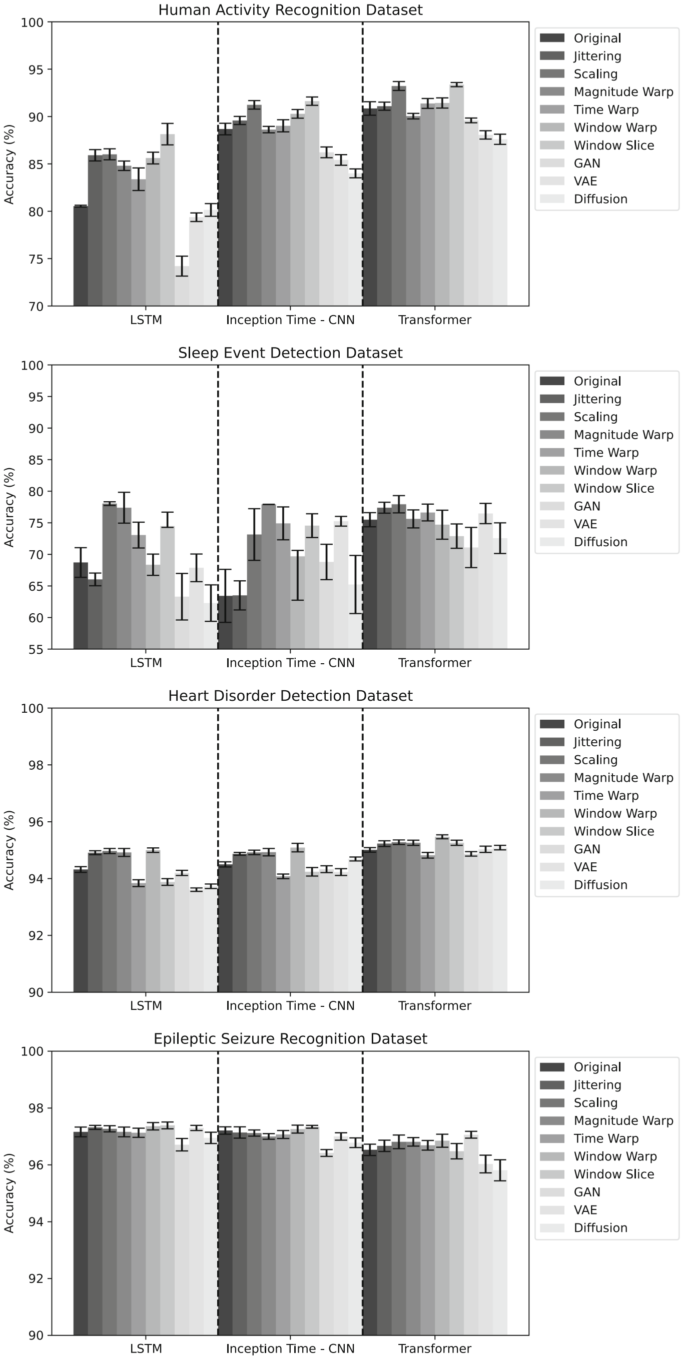


Fig. 3. Classification accuracy results for the combination of four datasets, three classifier architectures, and ten augmentation techniques explored in this study.

5 Discussion

Our investigation into data augmentation's impact on time series classification accuracy has elucidated several crucial insights. Notably, while augmentation typically boosts classification accuracy, the effectiveness of specific techniques varies depending on the dataset and classification architecture used.

Traditional augmentation methods, such as jittering and scaling, generally enhance model performance across various datasets by introducing necessary variability without significantly distorting the time series' inherent dynamics. However, the mixed results observed with window warping and slicing highlight the context-sensitive nature of augmentation effectiveness, indicating that a tailored approach, possibly involving a combination of techniques, might yield the best results.

The effectiveness of generative deep learning-based augmentation methods also varies. It appears that the addition of synthetic examples with class ratio distribution equal to the original dataset does not significantly boost the overall accuracy. However, the boost in performance may be more pronounced when the class distribution is imbalanced and synthetic examples are introduced to the minority class(es) to mitigate the class imbalance.

Furthermore, our findings reveal that multi-channel datasets tend to benefit more from augmentation than single-channel datasets, likely due to the richer information content that provides more scope for effective augmentation without loss of signal integrity. Conversely, the application of augmentation techniques, especially in datasets with low signal-to-noise ratios like EEG data, requires careful consideration to avoid degrading the classification accuracy.

Interestingly, the impact of augmentation appears to be relatively consistent across different classification architectures, indicating that the benefits of data augmentation transcend architectural differences and largely depend on the quality and diversity of the training data.

In summary, data augmentation emerges as a valuable tool for improving time series classification models, with its effectiveness highly contingent on the dataset characteristics, augmentation technique, and classification architecture. A judicious, context-aware application of augmentation techniques is essential to optimize model performance, highlighting the need for ongoing research to refine these strategies for diverse applications.

6 Conclusion and Future Work

Our comprehensive exploration of data augmentation strategies for time series classification in the biomedical domain has illuminated their varied impacts on model performance. We have shown that the effectiveness of augmentation techniques is highly context-specific, with no one-size-fits-all solution. This underscores the necessity for a tailored approach, informed by the dataset's characteristics and the model's requirements.

Moving forward, the development of more sophisticated, adaptive augmentation methods that can autonomously determine the most effective strategies for a given dataset and task is an exciting area for further exploration. While this study has focused on biomedical time series data, the insights gained are broadly applicable across various domains, pointing towards the broader goal of improving model robustness and generalization through strategic data augmentation.

Appendix

The tables below show the detailed classification accuracy results for each dataset, augmentation technique, and classification algorithm architecture. The LSTM, CNN, and Transformer table headers correspond to the LSTM-FCN [5], the Inception-Time CNN [3], and the TST [14] architectures respectively, as implemented in the TSAI library [11]. The "Par." table header indicates the tuned hyperparameter value used for that particular augmentation technique in the corresponding experiment. For example, a jittering value of 0.03 corresponds to the standard deviation value of the added Gaussian noise, a scaling value of 0.7 corresponds to the scaling factor multiplier, etc. For the generative models, the default hyperparameters recommended by the original model authors were used without tuning (Tables 1, 2, 3 and 4).

More implementation details can be found in the public source code page of the project: https://github.com/imics-lab/time-series-augmentation

Table 1. Results for Human Activity Recognition dataset.

Augmentation	Par.	LSTM	Par.	CNN	Par.	Transformer
Original	Null	80.54% ± 0.1	Null	88.68% ± 0.61	Null	90.85% ± 0.71
Jittering	0.03	85.91% ± 0.6	0.05	89.58% ± 0.43	0.8	91.09% ± 0.43
Scaling	0.7	86.01% ± 0.58	3	91.23% ± 0.45	3	93.22% ± 0.47
Mang. Warp.	0.1	84.8% ± 0.5	0.1	88.62% ± 0.34	0.3	90.04% ± 0.29
Time Warp.	0.1	83.38% ± 1.19	0.1	89.02% ± 0.64	0.1	91.38% ± 0.52
Window Warp.	0.01	85.62% ± 0.62	0.9	90.28% ± 0.46	0.01	91.43% ± 0.54
Window Slic.	0.9	88.14% ± 1.13	0.9	91.62% ± 0.44	0.9	93.37% ± 0.22
TTS GAN	Null	74.21% ± 1.05	Null	86.22% ± 0.57	Null	89.59% ± 0.25
VAE	Null	79.37% ± 0.45	Null	85.41% ± 0.56	Null	88.06% ± 0.44
Diffusion	Null	80.14% ± 0.67	Null	84% ± 0.47	Null	87.6% ± 0.54

Table 2. Results for Sleep Event Detection dataset.

Augmentation	Par.	LSTM	Par.	CNN	Par.	Transformer
Original	Null	68.72% ± 2.35	Null	63.43% ± 4.2	Null	75.49% ± 1.12
Jittering	0.01	66.04% ± 1	0.03	63.51% ± 2.3	0.07	77.39% ± 0.86
Scaling	11	78.04% ± 0.29	11	73.15% ± 4.09	9	77.94% ± 1.37
Magnitude Warping	3	77.39% ± 2.44	11	77.92% ± 0.02	9	75.62% ± 1.42
Time Warping	3	73.06% ± 2.04	11	74.91% ± 2.6	0.1	76.63% ± 1.33
Window Warping	0.9	68.36% ± 1.69	0.7	69.69% ± 3.94	0.09	74.7% ± 2.3
Window Slicing	0.2	75.48% ± 1.21	0.1	74.55% ±1.88	0.7	72.89% ± 1.93
TTS GAN	Null	63.3% ± 3.68	Null	68.8% ± 2.8	Null	71.09% ± 3.18
VAE	Null	67.87% ± 2.19	Null	75.25% ± 0.77	Null	76.47% ± 1.6
Diffusion	Null	62.29% ± 2.89	Null	65.23% ± 4.6	Null	72.57% ± 2.43

Table 3. Results for Heart Disorder Detection dataset.

Augmentation	Par.	LSTM	Par.	CNN	Par.	Transformer
Original	Null	94.32% ± 0.1	Null	94.5 % ± 0.09	Null	95.01% ± 0.08
Jittering	0.01	94.91% ± 0.07	0.01	94.87% ± 0.05	0.03	95.23% ± 0.1
Scaling	0.1	94.97% ± 0.09	0.1	94.92% ± 0.08	0.1	95.28% ± 0.08
Mang. Warp	0.1	94.92% ± 0.14	0.1	94.93% ± 0.13	0.1	95.26% ± 0.09
Time Warp	0.1	93.84% ± 0.12	0.1	94.08% ± 0.08	0.1	94.82% ± 0.1
Window Warp	0.01	95 % ± 0.08	0.03	95.09% ± 0.15	0.03	95.47% ± 0.07
Window Slic	0.01	93.88 % ± 0.12	0.01	94.24% ± 0.15	0.01	95.26% ± 0.09
TTS GAN	Null	94.2% ± 0.09	Null	94.33% ± 0.12	Null	94.87% ± 0.08
VAE	Null	93.61% ± 0.06	Null	94.23% ± 0.12	Null	95.03% ± 0.11
Diffusion	Null	93.73% ± 0.08	Null	94.69% ± 0.07	Null	95.09% ± 0.08

Table 4. Results for Epileptic Seizure Detection dataset.

Augmentation	Par.	LSTM	Par.	CNN	Par.	Transformer
Original	Null	97.16% ± 0.17	Null	97.21 % ± 0.13	Null	96.53% ± 0.2
Jittering	0.2	97.32% ± 0.07	0.4	97.14% ± 0.2	1	96.67% ± 0.2
Scaling	0.1	97.27% ± 0.11	0.1	97.12% ± 0.11	0.1	96.81% ± 0.24
Magnitude Warping	0.1	97.16% ± 0.17	0.1	97% ± 0.1	0.1	96.81% ± 0.15
Time Warping	7	97.13% ± 0.16	3	97.07% ± 0.14	0.1	96.69% ± 0.17
Window Warping	0.07	97.36 % ± 0.13	0.7	97.26% ± 0.14	0.1	96.85% ± 0.23
Window Slicing	0.3	97.39 % ± 0.12	0.3	97.34% ± 0.05	0.9	96.48% ± 0.27
TTS GAN	Null	96.71% ± 0.22	Null	96.42% ± 0.12	Null	97.06% ± 0.12
VAE	Null	97.3% ± 0.09	Null	97% ± 0.13	Null	96.03% ± 0.31
Diffusion	Null	96.95% ± 0.2	Null	96.78% ± 0.17	Null	95.81% ± 0.37

References

1. Barak, S., Mirafzali, E., Joshaghani, M.: Improving deep learning forecast using variational autoencoders. Available at SSRN 4009937 (2022)
2. Iglesias, G., Talavera, E., González-Prieto, Á., Mozo, A., Gómez-Canaval, S.: Data augmentation techniques in time series domain: a survey and taxonomy. Neural Comput. Appl. **35**(14), 10123–10145 (2023)
3. Ismail Fawaz, H., et al.: Inceptiontime: finding Alexnet for time series classification. Data Min. Knowl. Disc. **34**(6), 1936–1962 (2020)
4. Iwana, B.K., Uchida, S.: An empirical survey of data augmentation for time series classification with neural networks. PLOS ONE **16**(7), e0254841 (2021)
5. Karim, F., Majumdar, S., Darabi, H., Chen, S.: LSTM fully convolutional networks for time series classification. IEEE Access **6**, 1662–1669 (2017)
6. Korompili, G., et al.: PSG-audio, a scored polysomnography dataset with simultaneous audio recordings for sleep apnea studies. Scientific data **8**(1), 197 (2021)
7. Li, X., Metsis, V., Wang, H., Ngu, A.H.H.: TTS-GAN: a transformer-based time-series generative adversarial network. In: Michalowski, M., Abidi, S.S.R., Abidi, S. (eds.) AIME 2022. LNCS, vol. 13263, pp. 133–143. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-09342-5_13
8. Li, X., Sakevych, M., Atkinson, G., Metsis, V.: Biodiffusion: a versatile diffusion model for biomedical signal synthesis. arXiv preprint arXiv:2401.10282 (2024)
9. Micucci, D., Mobilio, M., Napoletano, P.: Unimib shar: a dataset for human activity recognition using acceleration data from smartphones. Appl. Sci. **7**(10), 1101 (2017)
10. Moody, G.B., Mark, R.G.: The impact of the MIT-BIH arrhythmia database. IEEE Eng. Med. Biol. Mag. **20**(3), 45–50 (2001)
11. Oguiza, I.: TSAI - a state-of-the-art deep learning library for time series and sequential data. Github (2023). https://github.com/timeseriesAI/tsai
12. Wen, Q., et al.: Time series data augmentation for deep learning: a survey. arXiv preprint arXiv:2002.12478 (2020)
13. Wu, Q., Fokoue, E.: Epileptic seizure recognition. UCI Machine Learning Repository (2017). https://doi.org/10.24432/C5G308
14. Zerveas, G., Jayaraman, S., Patel, D., Bhamidipaty, A., Eickhoff, C.: A transformer-based framework for multivariate time series representation learning. In: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, pp. 2114–2124 (2021)

The Impact of Synthetic Data on Fall Detection Application

Minakshi Debnath(✉), Md Shahriar Kabir, Jianyuan Ni,
and Anne Hee Hiong Ngu

Texas State University, San Marcos, TX, USA
{stg60,angu}@txstate.edu

Abstract. Lack of real-world data in clinical fields poses a major obstacle for training deep learning models. Using data augmentation can increase data volume, making the training of deep learning models more effective. This paper aims to investigate different techniques for generating realistic multivariate synthetic fall data, addressing the challenge of limited fall data availability. We experimented with three traditional time series data augmentation techniques, a generative AI approach with diffusion, and extraction of data from public video recordings of older adults falling. We evaluated the effectiveness of the generated data with both an LSTM model trained offline and using the SmartFall App running the LSTM model in real-time. Initial results indicate a 7–10% increase in the F1-score for the fall detection model when trained with additional data generated through the diffusion method during offline evaluation and a notable improvement of 24% was observed with the real-time evaluation of the model.

Keywords: Time series data generation · Fall detection · Diffusion model · Video data extraction

1 Introduction

Falling poses a significant health risk for older adults globally [9]. In fact, the injury posed by falling in older adults are the leading cause of unintentional death in individuals over 85 years old [15]. Research on wearable device technologies like smartwatches and IMU sensors for fall detection has become popular due to their affordability, portability, and non-intrusiveness. In complex physiological processes like fall onset, deep learning struggles with limited training data as fall events are rare and large data collection is difficult. Researchers have collected simulated fall data in controlled environments, a costly and labor-intensive process. Data augmentation or synthetic data generation techniques are one of the standard approaches to addressing the issue of small datasets [5]. Generative AI, like GANs, VAEs, and Diffusion Models, is prominent in creating synthetic data for images and time series. Diffusion models have become a popular method among deep generative models, showcasing outstanding performance in diverse applications [17]. More recently, virtual IMU signal has been reported as a reliable alternative way for synthetic data. For instance, an engineering pipeline was proposed

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 204–209, 2024.
https://doi.org/10.1007/978-3-031-66538-7_21

to generate on-body virtual sensor data utilizing data of a different modality (i.e., video) [6]. Therefore, we have adopted the methodology presented in [8] for the extraction of video fall data publicly available from two long-term care facilities in British Columbia [14]. In this work our contributions include: A) Introducing the Diffusion model for data generation. B) Extracting fall data from videos using pose estimation. C) Validating synthetic data techniques. D) Comparing fall detection model performance with real and synthetic data using the SmartFall App. E) Showing the effectiveness of data generated with the Diffusion model and video extraction in improving fall detection models.

2 Experimental Setup

Datasets: We employed three fall-based datasets as input to different synthetic data generation techniques and one video dataset for extraction of fall data for impact assessment. Those are SmartFallMM's smartwatch data (accelerometer data) (collected in our laboratory) [2], the UniMiB [11], and the K-Fall [18]. All those datasets have various simulated falls and activities of daily life performed by healthy young adults. The video dataset is a real-life video recording of older adults falling in a long-term care facility in British Columbia [14].

Data Preprocessing, Deep learning Model, Training and Evaluation: We used a basic LSTM deep learning model, which is favored for time series data due to its capability to learn temporal dynamics. The detail of the architecture can be found in our technical report [1]. Our model, deployed and tested in our SmartFall App, outperformed 1D CNN, Gradient Boosting, and Random Forest [10].

The input data is pre-processed by segmenting into overlapping windows with a step size of 10, using a window size of 128 across all experiments. Different training scenarios are explored, with baseline models trained solely on original datasets, without any synthetic data. The dataset is split into training, validation, and test sets at a ratio of 70/20/10, and a 5-fold validation method is applied. Baseline models serve as the reference. New LSTM models are trained using combined original and synthetic data, while validation and testing are conducted solely on real data. Performance evaluation includes standard metrics: Precision, Recall, F1-score, and Accuracy, to assess the effectiveness of synthetic data from various methods.

We validate the best model using generated data with the SmartFallMM dataset in a real-world setting via the SmartFall App [12]. Three students participated in the evaluation under IRB 7846 at Texas State University. They wore watches on the left wrist with the SmartFall App installed, executing falls on an air mattress and daily activities. Both correct and incorrect predictions were recorded.

3 Synthetic Data Generation

Basic Data Augmentations: We employed three data augmentation techniques, namely Jittering [13], Magnitude Warping [13], and Rotation [16]. Jittering involves augmenting time series data with random Gaussian noise. Magnitude warping is a technique applied to time series datasets where the magnitude of each sample is modified. This modification is achieved by multiplying the original time series with a cubic spline curve. The rotation augmentation technique serves as a means to simulate various sensor placements (e.g. left vs right wrist), introducing the diversity of data patterns without modifying the inherent labels associated with the data.

Diffusion Method: Denoising Diffusion Probabilistic Models (DDPMs) represent a class of generative AI models that have demonstrated remarkable success in synthesizing high-quality data across domains such as images and audio [4, 7]. We have integrated diffusion models with a U-Net architecture adapted from previous work [7]. Originally designed for image analysis, this U-Net architecture has been reconfigured for time-series data using one-dimensional (1D) convolutional layers with a kernel size of 7 and padding of 3, capturing essential temporal dependencies in time series data. Figure 1 represents the architecture used for this work.

Upon receiving the time-series input, the data undergoes normalization with RMSNorm, which stabilizes the training process. The network architecture, comprising ResNet blocks and Linear Attention units, executes down sampling and up sampling operations to refine features and preserve temporal information. Time and sinusoidal positional embeddings are integrated within each block, ensuring the model's responsiveness to the diffusion timesteps and sequence positions. Li's original model [7] posed challenges in handling variable-length accelerometer data for different types of falls and lacked a stable normalization technique. We improved the model by incorporating a padding strategy during preprocessing, enabling consistent input size, and avoiding loss of information. Details of the method can be found in our technical report [1].

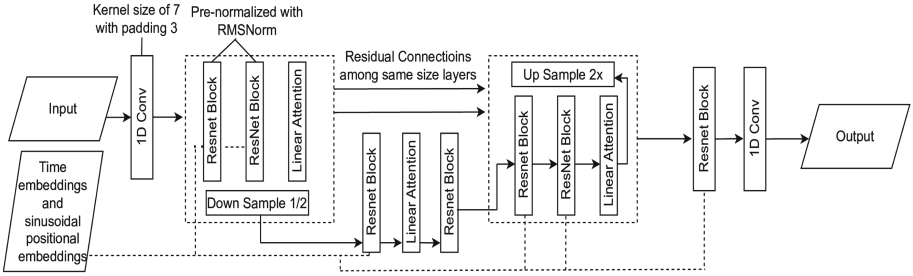


Fig. 1. Schematic of the U-Net Architecture Adapted for Time-Series Data.

Extraction of fall data from video via Pose Estimation: We have adopted the methodology presented in [8] for the extraction of video fall data. To extract the fall data correctly, we edited 34 publicly available videos sourced from [14]. We first isolated the falling person in the video by cropping the frame around them to reduce the time for the extraction process and to zoom in on the most relevant data to extract. We ensure to include 1 to 2 s of pre-fall and post-fall segments. Resolution and brightness adjustments are made for each video. The 3D pose estimation extracted 17 joint positions from each video's detected human skeleton. For generating synthetic data, if we aim to add video fall data to the SmartFallMM dataset, we focus on extracting accelerometer data from the left wrist joint position. Alternatively, for UniMiB, we extract accelerometer data from the left and right hips' joint positions. If we are creating synthetic data for UniMiB, we will extract accelerometer data from the left and right hips' joint positions. After pose estimation, we use 3D key points to extract acceleration data. Calculating velocity from position changes, then acceleration from velocity changes, we extract about 30 fall samples. Figure 2 outlines this methodology for deriving accelerometer readings from a video capturing an elderly person's fall.

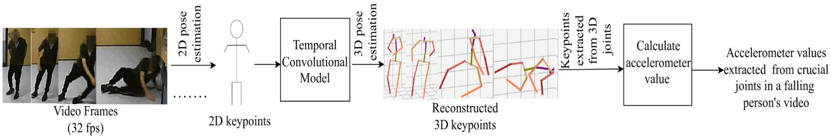

Video Frames
(32 fps)
2D pose estimation
....
2D keypoints
Temporal Convolutional Model
3D pose estimation
Reconstructed 3D keypoints
Keypoints extracted from 3D joints
Calculate accelerometer value
Accelerometer values extracted from crucial joints in a falling person's video

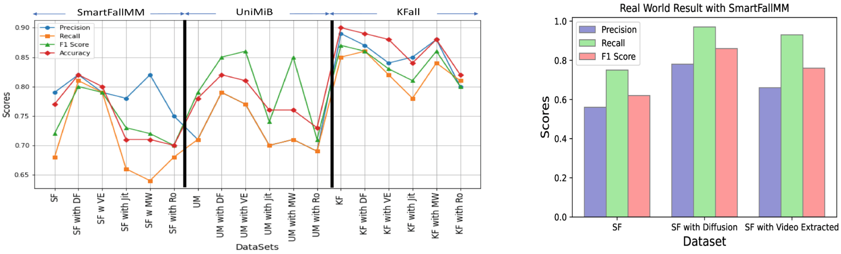
Fig. 2. Accelerometer data extraction process from video frames.

4 Results

Offline Evaluation of Fall Detection Model: Figure 3(a) compares datasets and synthetic data using three methods. Results from 5-fold validation, including precision, recall, F1-score, and accuracy, are shown for each. Each colored line represents the variation of each metric across different datasets. Abbreviations SF, UM, and KF represent SmartFallMM, UniMiB, and K-Fall datasets, while DF, VE, Jit, MW, and Ro stand for Diffusion, Video Extraction, Jittering, Magnitude Warping, and Rotation. Only SmartFallMM dataset (SF) achieves F1 score of 0.72 and accuracy of 0.77. With synthetic data, especially using diffusion, SF's F1-score improves to 0.80, nearly 10% better. Pose estimation-based data extraction also boosts performance. We additionally assessed and compared results across two other public datasets, UniMiB and K-Fall. The baseline F1 score and accuracy for UniMiB (UM) are 0.79 and 0.78, respectively. We noted a enhancement in performance by incorporating synthetic data generated via the diffusion method. The F1 score increased from 0.79 to 0.85, reflecting an improvement rate of nearly 7%. Despite incorporating diffusion-generated and video-extracted data, there was no improvement observed for K-Fall (KF). This could be attributed to the larger size of the K-Fall dataset compared to the other two datasets, the added data does not lead to more generalization with the simple LSTM architecture.

The better performance gap of synthetic data from diffusion and pose extraction methods as compared to basic augmentation likely stems from the quality of the added information. Data from diffusion and pose extraction enriches the dataset with meaningful patterns and the generated data aligns better with real data.

Real-time Evaluation of Fall Detection Model: Figure 3(b) showcases the real-time evaluation result for the top-performing offline model. We only tested the offline model with SmartFallMM watch data because our SmartFall app exclusively uses watch-sensed data. We share results from testing the SmartFall App across three participants, starting with an initial F1 score of 0.62 using basic LSTM model for SF. Next, we evaluated



(a) (b)

Fig. 3. (a) Evaluation of SmartafallMM, UniMiB, and K-Fall with Synthetic Data Generated using three different methods. (b) Real world result with SmartFallMM.

top models trained with a mix of synthetic and real data: SF with Diffusion and SF with Video Extracted. The top SmartFall App model, trained with diffusion-generated data, achieved an F1 score of 0.86 (24% improvement), while the video-extracted data model reached 0.76 (14% improvement). Real-time testing confirms synthetic data's effectiveness in enhancing fall detection methods.

5 Discussion and Future Work

This study explores methods to generate synthetic fall data to overcome data scarcity. Enhanced performance is observed in offline evaluation for SmartFallMM and UniMiB with diffusion and video-extracted synthetic fall data. Additionally, promising real-time performance is demonstrated for SmartFallMM with synthetic data from diffusion and video extraction. In the future, we aim to identify the ideal balance of real and synthetic data for training robust models, alongside exploring video extraction methods via AI platforms like Sora [3], which generate videos from textual descriptions.

Acknowledgment. We thank the National Science Foundation for funding the research under the NSF-SCH (21223749).

References

1. Enhancing fall detection: The role of synthetic data. https://drive.google.com/file/d/1WLcxcjwg1d_t1i0T2RLr930Cpx-j64tt/view
2. Smartfallmm watch accelerometer dataset. https://drive.google.com/file/d/10tOrG7zgbLOgBJTFj0PDl3DYFjz7kkTO/view
3. Sora. https://openai.com/sora
4. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M.F., Lin, H. (eds.) NeurIPS, vol. 33, pp. 6840–6851. Curran Associates, Inc. (2020)
5. Iwana, B.K., Uchida, S.: An empirical survey of data augmentation for time series classification with neural networks. PLoS ONE **16**(7), e0254841 (2021)

6. Kwon, H., et al.: Imutube: Automatic extraction of virtual on-body accelerometry from video for human activity recognition. IMWUT **4**(3), 1–29 (2020)
7. Li, X.: Mitigating data shortage in biomedical signal analysis: anbinvestigation intobtransfer learning and generative models. Ph.D. dissertation, Texas State University, Texas, June 2023
8. Liu, J., et al.: A graph attention spatio-temporal convolutional network for 3D human pose estimation in video. In: ICRA, pp. 3374–3380. IEEE (2021)
9. Matos-Carvalho, J.P., Correia, S.D., Tomic, S.: Sensitivity analysis of lstm networks for fall detection wearable sensors. In: CIoT, pp. 112–118 (2023)
10. Mauldin, T.R., Ngu, A.H., Metsis, V., Canby, M.E.: Ensemble deep learning on wearables using small datasets. ACM Trans. Comput. Healthcare **2**(1), 1–30 (2021)
11. Micucci, D., Mobilio, M., Napoletano, P.: UniMiB SHAR: a dataset for human activity recognition using acceleration data from smartphones. Appl. Sci. **7**(10), 1101 (2017)
12. Ngu, A.H., Yasmin, A., Mahmud, T., Mahmood, A., Sheng, Q.Z.: Demo: P-fall: personalization pipeline for fall detection. In: Proceedings of the 8th ACM/IEEE International Conference on Connected Health: Applications, Systems and Engineering Technologies, CHASE 2023, pp. 173–174. Association for Computing Machinery, New York (2024)
13. Nikitin, A., Iannucci, L., Kaski, S.: TSGM — a flexible framework for synthetic time series generative modeling (2023)
14. Robinovitch, S.: Falls experienced by older adult residents in long-term care homes (2018)
15. Zebiah, S.S. et al.: Human fall detection using machine learning and deep learning techniques: a survey. In: ICSPC, pp. 253–257 (2023)
16. Um, T.T., et al.: Data augmentation of wearable sensor data for Parkinson's disease monitoring using convolutional neural networks. In: ICMI, pp. 216–220. ACM, New York (2017)
17. Yang, L., et al.: Diffusion models: a comprehensive survey of methods and applications. ACM Comput. Surv. **56**(4), 1–39 (2023)
18. Xiaoqun, Y., Jang, J., Xiong, S.: A large-scale open motion dataset (kfall) and benchmark algorithms for detecting pre-impact fall of the elderly using wearable inertial sensors. Front. Aging Neurosci. **13**, 692865 (2021)

Natural Language Processing

A Retrieval-Augmented Generation Strategy to Enhance Medical Chatbot Reliability

Saba Ghanbari Haez[1,2], Marina Segala[1], Patrizio Bellan[1], Simone Magnolini[1], Leonardo Sanna[1(✉)], Monica Consolandi[1], and Mauro Dragoni[1]

[1] Fondazione Bruno Kessler, Trento, Italy
{sghanbarihaez,msegala,pbellan,magnolini,lsanna,mconsolandi,dragoni}@fbk.eu

[2] Free University of Bozen, Bolzano, Italy

Abstract. The advent of Large Language Models opened new perspectives concerning their usage within the digital health domain. However, their intrinsic probabilistic and unpredictable behavior needs the design of trustworthy strategies aiming to avoid the creation of hallucinations that, especially within the digital health domain, may lead to severe harm. Such an issue has been addressed with the adoption of Retrieval-Augmented Generation solutions, where the text generation task is supported by controlled knowledge injected into the prompts. Even if the hallucination issue is mitigated, the generation of certified information (such as trustworthy content granted by the system's owner) requires more sophisticated strategies. In this work, we propose an approach where the classic Retrieval-Augmented Generation pipeline is enhanced with a further initial step where the Large Language Model is asked to generate a preliminary text used to query the repository of certified information for presenting the appropriate content to the final user.

1 Introduction

Large Language Models (LLMs) such as BERT [3] and T5 [18] possess the ability to generate factual information based on learned patterns from extensive training data [16]. However, their accuracy without external sources may vary due to several factors like data quality, task complexity, and parameter density. Therefore, they may generate inaccurate or fictional content, i.e., hallucinations [23,27]. Recent efforts aim to tackle these challenges by augmenting external knowledge to empower LLMs to interact effectively with users and their surroundings.

Retrieval-Augmented Generation (RAG) [11], explicitly incorporates external knowledge into LLMs' prompts to contribute to the enhancement of their trustworthiness [6]. This involves retrieving documents relevant to the user's query and subsequently generating a comprehensive response considering the contained factual information. The efficiency of RAG systems relies on sufficient and diverse training data, with the risk of observing a low accuracy if

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 213–223, 2024.
https://doi.org/10.1007/978-3-031-66538-7_22

the retrieval system lacks robustness and reliability. In particular, the *retrieval* phase struggles with issues like semantic ambiguity coupled with basic matching techniques that lead to inaccurate data retrieval. While, the *augmentation* and *generation* phases struggle with integrating context and coherence, resulting in superficial responses that fail to meet sophisticated query demands[1].

Previous research predominantly employs traditional RAG pipelines, which involve retrieving documents relevant to user input from a non-certified database before sending the user query to the LLM. The novel contribution of this paper lies in (i) using a certified document repository to inject factual knowledge into the LLM; and, (ii) proposing a novel enhancement to the classic RAG framework by introducing a further preliminary interaction with the integrated LLM[2].

We present the theoretical framework and discuss its preliminary adoption within an FAQ-based chatbot designed to support pregnant women followed by the Trentino Healthcare Department in Italy, i.e., the *TreC-Mamma* application[3]. Our goal is to answer the following Research Questions: (RQ1) How do the LLMs be integrated effectively into digital health solutions? (RQ2) Can LLMs generate *certified* content (the meaning of the term "certified" is explained in Sect. 3)? (RQ3) Can a RAG strategy be enhanced to solve semantic ambiguity and avoid hallucinations?

Through the empirical evaluation discussed in Sect. 4, we demonstrate the effectiveness of the proposed RAG-based strategy in addressing the limitations of LLMs and enhancing the overall user experience in domains targeted by the current work, i.e., maternal health. Moreover, we suggest that our methodology may represent a promising direction for leveraging advanced language technologies to tackle the certified information challenges effectively.

2 Related Work

In recent years, there has been an increase in research efforts to improve the credibility and effectiveness of LLMs, particularly in domains that prioritize accuracy and reliability, such as modern medicine and digital health [17]. Conversational Artificial Intelligence (AI) in healthcare suffers several challenges that are crucial to address, including the lack of suitable evaluation metrics, concerns regarding fairness, bias, and hallucination in chatbot responses, the balance between personalization and oversimplification, and obstacles in implementation [1]. In this context, RAG demonstrated to be a suitable candidate for mitigating hallucination issues, enriching factual content generation in LLMs, and integrating external knowledge sources. We believe these advancements can enhance the utility of conversational AI in healthcare settings.

RAG has recently gained attention for its explicit incorporation of external knowledge into LLMs' prompts. By leveraging Information Retrieval (IR) techniques, RAG aims to enhance the credibility and relevance of generated content

[1] https://arxiv.org/abs/2401.05856.

[2] In this work, we adopted the GPT4 LLM.

[3] https://trentinosalutedigitale.com/blog/portfolio/trec-mamma/.

by retrieving documents pertinent to user queries. RAG utilizes both parametric and non-parametric memory, drawing upon pre-trained seq2seq models [11] and Dense Passage Retrieval (DPR) [10]. This approach has led to surpassing state-of-the-art performance on tasks like QA and summarization and has been noted for enhancing language generation by producing more specific, diverse, and factual language compared to parametric-only seq2seq models [22]. Nevertheless, while RAG has shown promise in improving the output of LLMs, it faces constraints when confronted with data outside its training set. Several approaches have been explored to address this issue.

Building upon the works by [10,11], Retrieval-Augmented Language Model pre-training (REALM) [6] integrates a knowledge retrieval mechanism to enhance neural language models' performance in question-answering tasks, demonstrating superior accuracy and interoperability. A two-stage Approach proposed by [8] combines DPR with generative sequence-to-sequence language models, leveraging the strengths of both approaches to provide comprehensive and contextually relevant responses for open-domain QA tasks.

I-RetGen (Iterative Retrieval-Generation) [21] iteratively integrates retrieval and generation processes, enhancing relevance for complex queries while minimizing overhead by utilizing the LLM's generation output to guide retrieval. RAGE (Retrieval-Augmented Generation with Rich Answer Encoding) [7] combines retrieval and generation techniques to produce informative and coherent answers, enhancing the richness and relevance of generated content. FLARE (Active Retrieval-Augmented Generation) [9] dynamically decides when and what to retrieve throughout the generation process, offering a proactive approach to content augmentation.

Recent advancements have further expanded the capabilities of retrieval-augmented generation, including techniques such as Augmentation-Adapted Retriever (AAR) [28] and Knowledge-Augmented Language Model Verification (KALMV) [2]. These approaches aim to enhance language model accuracy across different domains by integrating external knowledge and detecting errors in both knowledge retrieval and text generation processes. Additionally, frameworks like Induction-Augmented Generation (IAG) [29] and domain adaptation techniques for RAG models [22] demonstrate ongoing efforts to improve implicit reasoning and adaptability in question-answering tasks.

Despite the diversity of approaches, prior studies have concentrated on traditional RAG pipelines, which focus exclusively on enhancing text generation. Our proposed approach differs significantly by involving instead the use of a certified repository as well as the injection of knowledge directly into the LLM. By reconfiguring the RAG process and integrating retrieved information into the answer-generation pipeline, we aim not only to mitigate hallucination issues but also to ensure the generation of certified and contextually informed responses. This novel methodology represents a significant advancement in addressing the challenges of factual content generation within LLM frameworks, particularly in sensitive domains like digital health. Furthermore, our approach holds promise for improving user trust and satisfaction in automated systems by providing reliable and contextually relevant information. Additionally, our empirical evaluation and comparative analysis demonstrate the effectiveness of our method in enhancing the user experience, particularly in domains such as maternal nutrition and health, as evidenced by our preliminary adoption of the TreC Mamma application.

3 Method

This Section presents a preliminary discussion about how RAG may be a suitable strategy to mitigate the hallucination issues affecting LLMs followed by a description of how we implemented our strategy.

3.1 Preliminaries

As introduced in Sect. 1, the variability of LLMs' responses poses a substantial issue when operating in contexts where certified information is needed. In this work the term "*certified information*" refers to text created or verified by healthcare professionals, ensuring it aligns with current scientific knowledge in the specific domain. To preserve the nature of "*certified information*", it is essential that the content remains unchanged in its textual form. Moreover, it should be semantically predetermined, i.e., each specific question consistently corresponds to a particular set of semantic equivalent answers. Using LLMs, even with RAG, cannot guarantee this requirement.

Indeed, a standard RAG pipeline in a FAQ-based chatbot would employ the user's question to query the certified documents. Usually, RAG converts a user query into a vector embedding representation, which is then employed to evaluate semantic similarity across the document repository. Yet, there can be significant differences between the vector representations of the query and the documents within the semantic space. This disparity poses a notable limitation as it could result in relevant documents being overlooked during retrieval.

Moreover, RAG provides additional opacity to the algorithm since the user does not know which information is being used to provide the answer. Although our conversational agent is not directly involved in diagnostic processes, some ethical questions about possible bias of LLMs remain valid [24] as well as the need to build a system as much as possible transparent and accountable [25]. Given the intrinsic opacity of LLMs, it is nonetheless our effort to pursue "*explicability*", i.e., the new ethical principle that Floridi et al. [4] introduced alongside the traditional ones (beneficence, non-maleficence, autonomy, and justice). This ethical principle is already largely used in the field of explainable AI [20]. Hence, in our chatbot, we want to provide the sources used in the RAG process to ensure transparency at the epistemic level [26]. This strategy aims to prevent possible trust issues towards the agent [14] or at least to mitigate them.

3.2 Implementation

The solution we propose in this paper is summarized in Fig. 1.

Our goal is to address the limitations mentioned above with a modular approach aiming to enhance the classic RAG pipeline with the integration of the Hypothetical Document Embeddings (HyDE) framework [5]. The HyDE introduces a further step at the beginning of the pipeline where the LLM is asked to produce a hypothetical document (HyDoc) based only on the input query provided by the user. The hypothetical document represents the query's information request and is also meant to capture relevant textual patterns that might be present in the certified repository connected to the pipeline. It is important to mention that, at this stage, the output generated by the LLM model might contain hallucinations since any check is performed. However, the HyDoc generated should lie in the semantic space in a neighborhood of similar real documents that contain the correct and certified answer to provide to the user.

Thus, the main idea is to use the HyDoc generated by the LLM to augment the initial query. To do that, we need to transform the HyDoc into a semantic vector, namely a sentence embedding. Creating an embedding representation of a sentence is challenging because the meaning of each word needs to be contextualized concerning the other words in the sentence. Anyhow, many LLMs are trained to predict the next word in a sequence, so the embedding representation of a sentence cannot be easily extracted. Indeed, unlike universal word embeddings methods, e.g., word2vec [15], a widely accepted general-purpose sentence embedding technique is still a very active research field [13].

In our work, we integrated the *paraphrase-multilingual-mpnet-base-v2* Bi-Encoder model [19] and we used it to both create the embeddings of our HyDoc

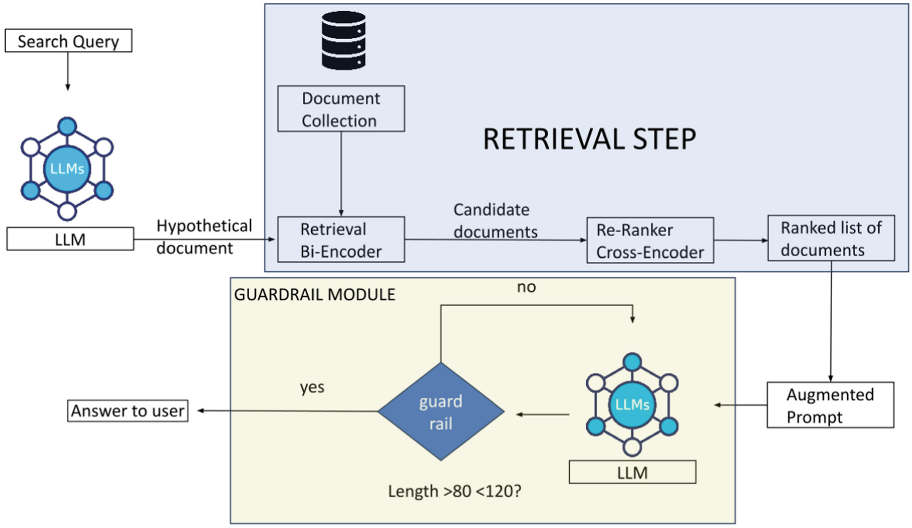


Fig. 1. Our RAG pipeline

and the vector-based representation of the whole repository. The model works by adding a pooling operation, which generates a fixed-size embedding representation normalized to have a total length of 1.00. Such a representation eases the comparison of each vector by adopting cosine similarity. After calculating the cosine similarity between all pairs of type $<HyDoc, D_i>$ with $i = 1...n$, where n is the number of documents contained in the certified repository and D_i the i_{th} document contained in the certified repository, we ranked the documents and selected the $k = 50$ most similar ones. The rationale behind the choice of $k = 50$ is that from an information retrieval perspective, this is an acceptable number of documents that may grant the inclusion of the most relevant ones [12].

The Bi-Encoder is a computationally efficient method for semantic search, but it works well only when we have documents of comparable lengths. Hence, if the HyDoc is either significantly shorter or longer than the documents to retrieve, the risk of retrieving non-relevant documents is considerable.

For this reason, we integrated a Cross-Encoder module to re-rank the list of retrieved documents. This post-retrieval operation ensures the selection of the most informative documents. For this task, we use the *ms-marco-MiniLM-L-6-v2*[4] cross encoder. The Cross-Encoder is thus more accurate than the Bi-Encoder although it is computationally more expensive. For this reason, we applied it only to the list of candidate relevant documents to reduce the overall computation time of each user's request aiming to preserve the usability of the system.

Once the re-ranking process is completed, we select the top $j = 3$ documents that are most similar to the HyDoc, according to the Cross-Encoder output. These three documents are then used to augment the original prompt and retrieve the textual part of the final answer sent to the user. Here, a *Guard-Rail* module[5] is applied to ensure that the reply generated by the LLM satisfies the length requested through the prompt. The response of the agent will therefore contain the generated text as well as the pointers to the original certified sources, i.e., the three selected documents, used to generate the answer.

4 Evaluation and Discussion

For this study, we have curated a document repository certified by the Trentino Healthcare Department. This dataset forms the backbone of the strategy we proposed in this work, by ensuring that the information fed into the integrated LLM is both accurate and authoritative. The primary source of our dataset comes from the Obstetrician Department of the Hospital of Trento. This includes a comprehensive collection of 1512 documents split as follows. A set of 179 informative cards associated with each pregnancy week offering information pertinent to maternal health, pregnancy, and fetus status. These cards have been written and certified by healthcare professionals, providing a reliable foundation for our model. In addition to the informative cards, this set contains additional content

[4] https://huggingface.co/cross-encoder/ms-marco-MiniLM-L-6-v2.

[5] For a detailed description of how this strategy works see the paper of Mangaokar et al. https://arxiv.org/abs/2402.15911.

extracted from videos released by the same department. Then, documents from two certified repositories: UPPA (953 documents) and ISS-Salute (380 documents) have been incorporated. Both sources have been considered *certified* by our healthcare department for their evidence-based approach to parenting and child health. The inclusion of both repositories introduces a broader perspective on child care, complementing the information provided by the Obstetrician Department of the Hospital of Trento. This amalgamation of resources from both the hospital and repositories ensures a comprehensive and well-rounded dataset that covers the spectrum of maternal and child health.

To address the research questions provided in Sect. 1, our system combines the generative capabilities of LLMs with a retrieval mechanism performed on the repository of certified documents. This hybrid approach ensures that the generated responses are not only linguistically coherent and contextually relevant but also grounded in verified medical knowledge.

To rigorously assess the performance of our solutions, we devised a comprehensive evaluation strategy that encompasses both the accuracy of the answers provided and the quality of the supporting documents retrieved from the certified repository. This dual-focus evaluation is crucial for ensuring that the solution delivers precise information and enriches its responses with credible and authoritative sources, thereby enhancing the trustworthiness and reliability of the system.

The evaluation task consisted of a set of 100 questions, which were considered representative by the experts involved in the evaluation process and that represent typical inquiries made by new mothers regarding pregnancy and early childcare. These questions were designed to cover a broad spectrum of topics within the domain, ensuring a thorough evaluation of the system's capabilities. The questions were then presented to a group of five test users, who interacted with the TreC Mamma application and collected the answers. The users were instructed to evaluate the responses based on seven criteria: (i) the relevance of the answer to the question, (ii) the relevance of the links (documents) provided, (iii) text quality, (iv) reliability, (v) clarity, (vi) completeness, and (vii) an overall evaluation score. These metrics were chosen to provide a holistic view of the system's performance, encompassing both the quality of the generated text and the relevance and certified documents provided within the answers.

The results of the evaluation are summarized in Table 1, which presents the average scores across all test users for each evaluation criterion. The first column contains the name of each metric; the second column contains the average score computed by considering the judgments provided by each user on all questions; the third and fourth columns contain the highest and the lowest scores obtained for that metric, respectively; and, the fifth column contains the variance. The metric [M1] involves binary classification (relevant, not relevant), and its score is interpretable as a percentage. The metric [M2] is a three-way classification (on-topic, partially on-topic, off-topic) and can similarly be interpreted as a percentage. The metrics from [M3] to [M7] employ a 5-level evaluation scale ranging from 1 (insufficient) to 5 (great).

Table 1. Summary of the results provided by the test users.

Evaluation Criterion	Avg	Max	Min	Var
[M1] Is the answer relevant to the question?	0.93	1.00	0.50	0.02
[M2] Links relevance	0.44	1.00	0.00	0.05
[M3] Text quality	4.59	5.00	3.33	0.06
[M4] Reliability	3.79	4.75	2.33	0.40
[M5] Clarity	4.60	5.00	3.33	0.05
[M6] Completeness	3.38	4.75	1.33	0.81
[M7] Overall evaluation	3.40	4.75	1.67	0.59

The high relevance score ([M1]), i.e., 0.93, indicates that the chatbot is highly effective in providing answers that are pertinent to the users' questions. However, the relevance of the links provided ([M2]), i.e., 0.44 suggests that there is significant room for improvement in the selection and presentation of supporting documents. The text quality, clarity, and reliability scores are relatively high, demonstrating the system's ability to generate well-written, clear, and somewhat reliable responses. Completeness and overall evaluation scores, while above average, highlight areas where further enhancements could be made to improve user satisfaction and the comprehensiveness of the answers provided. By considering the variance values, we may observe how the criteria [M4], [M6], and [M7] required further investigations. A preliminary further analysis revealed how, for some of the queries contained within the test set, the final output produced by the LLM did not satisfy the expectations of the evaluators.

As a final consideration, we state that we can positively answer the three research questions presented in Sect. 1. We can positively answer **RQ1** since the average score observed for the criterion [M7] proves an effective behavior of the proposed solution within the digital health domain. We can positively answer to **RQ2** as well, since the high values obtained for metrics [M1], [M3], and [M4] demonstrate how the content generated by the LLM can be considered certified. Finally, we can also positively answer to **RQ3** given the high values obtained for metrics [M2], [M4], [M5], and [M6] showing how, on average, the content of the final text sent to the evaluators has been considered correct, i.e., no hallucinations were included. The only point of attention related to the metric [M2] whose value demonstrates that there is still room for improvement, even if, on average, half of the documents included in the links sent to the users have been considered fully relevant.

5 Conclusions

In this work, we presented a framework showing how the RAG pipeline can be enhanced by introducing a further interaction with the integrated LLM before the retrieval step to support scenarios where the answer provided to users must contain only certified information. We tested our approach in the context of the local project TreC-Mamma, promoted by the Trentino Healthcare Department, which includes a mobile application used by pregnant women with an FAQ facility. Preliminary results demonstrated the suitability of the proposed strategy and paved the way for further steps in this research direction and future implementation in other domains. In particular, the effort will focus on the main limitation we observed, i.e., the retrieval module. Such a module is in charge of retrieving the certified information and, in the current setting, registered the lowest score compared with the other criteria adopted to evaluate the performance of the proposed solution. Finally, we intend to explore the integration of open LLMs that may represent a strong requirement concerning the deployment of this type of solution into production environments.

Acknowledgments. We acknowledge the support provided by the PNRR initiatives: INEST (Interconnected North-East Innovation Ecosystem), project code ECS00000043, and FAIR (Future AI Research), project code PE00000013. These projects are part of the NRRP MUR program, funded by the NextGenerationEU. This paper is supported by the TrustAlert project, funded by Fondazione Compagnia San Paolo and Fondazione CDP under the "Artificial Intelligence" call.

References

1. Abbasian, M., et al.: Foundation metrics for evaluating effectiveness of healthcare conversations powered by generative AI. NPJ Digit. Med. **7**(1), 82 (2024). https://doi.org/10.1038/s41746-024-01074-z
2. Baek, J., Jeong, S., Kang, M., Park, J., Hwang, S.: Knowledge-augmented language model verification. In: Bouamor, H., Pino, J., Bali, K. (eds.) Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, pp. 1720–1736. Association for Computational Linguistics, Singapore (2023). https://doi.org/10.18653/v1/2023.emnlp-main.107, https://aclanthology.org/2023.emnlp-main.107
3. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of deep bidirectional transformers for language understanding. In: Burstein, J., Doran, C., Solorio, T. (eds.) Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pp. 4171–4186. Association for Computational Linguistics, Minneapolis, Minnesota (2019). https://doi.org/10.18653/v1/N19-1423, https://aclanthology.org/N19-1423
4. Floridi, L., et al.: AI4people-an ethical framework for a good AI society: opportunities, risks, principles, and recommendations. Mind. Mach. **28**, 689–707 (2018)
5. Gao, L., Ma, X., Lin, J., Callan, J.: Precise zero-shot dense retrieval without relevance labels. In: Rogers, A., Boyd-Graber, J.L., Okazaki, N. (eds.) Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023. pp. 1762–1777. Association for Computational Linguistics (2023). https://doi.org/10.18653/V1/2023.ACL-LONG.99
6. Guu, K., Lee, K., Tung, Z., Pasupat, P., Chang, M.: Retrieval augmented language model pre-training. In: III, H.D., Singh, A. (eds.) Proceedings of the 37th

International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 119, pp. 3929–3938. PMLR (13–18 Jul 2020). https://proceedings.mlr.press/v119/guu20a.html

7. Huang, W., Lapata, M., Vougiouklis, P., Papasarantopoulos, N., Pan, J.Z.: Retrieval augmented generation with rich answer encoding. In: Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 1012–1025. Association for Computational Linguistics (2023)
8. Izacard, G., Grave, E.: Leveraging passage retrieval with generative models for open domain question answering. In: Merlo, P., Tiedemann, J., Tsarfaty, R. (eds.) Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume. pp. 874–880. Association for Computational Linguistics, Online (2021). https://doi.org/10.18653/v1/2021.eacl-main.74, https://aclanthology.org/2021.eacl-main.74
9. Jiang, Z., et al.: Active retrieval augmented generation. In: Bouamor, H., Pino, J., Bali, K. (eds.) Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, pp. 7969–7992. Association for Computational Linguistics, Singapore (2023). https://doi.org/10.18653/v1/2023.emnlp-main.495, https://aclanthology.org/2023.emnlp-main.495
10. Karpukhin, V., et al.: Dense passage retrieval for open-domain question answering. In: Webber, B., Cohn, T., He, Y., Liu, Y. (eds.) Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 6769–6781. Association for Computational Linguistics (2020). https://doi.org/10.18653/v1/2020.emnlp-main.550, https://aclanthology.org/2020.emnlp-main.550
11. Lewis, P., et al.: Retrieval-augmented generation for knowledge-intensive nlp tasks: Adv. Neural. Inf. Process. Syst. **33**, 9459–9474 (2020)
12. Li, H.: Learning to rank for information retrieval and natural language processing. Springer Nature (2022)
13. Li, R., Zhao, X., Moens, M.: A brief overview of universal sentence representation methods: a linguistic view. ACM Comput. Surv. **55**(3), 1–42 (2023). https://doi.org/10.1145/3482853
14. Martens, M., De Wolf, R., De Marez, L.: Trust in algorithmic decision-making systems in health: a comparison between ADA health and IBM Watson oncology. Cyberpsychology **18**(1) (2024)
15. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. Adv. Neural Inform. Proc. Syst. **26** (2013)
16. Petroni, F., et al.: Language models as knowledge bases? (2019) arXiv preprint arXiv:1909.01066
17. Pham, K.T., Nabizadeh, A., Selek, S.: Artificial intelligence and chatbots in psychiatry. Psychiatr. Q. **93**, 249–253 (2022). https://doi.org/10.1007/s11126-022-09973-8received 26 September 2021, Revised 23 January 2022, Accepted 26 January 2022, Published 25 February 2022
18. Raffel, C., et al.: Exploring the limits of transfer learning with a unified text-to-text transformer. J. Mach. Learn. Res. **21**(1), 5485–5551 (2020)
19. Reimers, N., Gurevych, I.: Sentence-bert: Sentence embeddings using siamese bert-networks. In: Inui, K., Jiang, J., Ng, V., Wan, X. (eds.) Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP

2019, Hong Kong, China, November 3-7, 2019. pp. 3980–3990. Association for Computational Linguistics (2019). https://doi.org/10.18653/V1/D19-1410

20. Saeed, W., Omlin, C.: Explainable AI (XAI): a systematic meta-survey of current challenges and future opportunities. Knowl.-Based Syst. **263**, 110273 (2023)
21. Shao, Z., Gong, Y., Shen, Y., Huang, M., Duan, N., Chen, W.: Enhancing retrieval-augmented large language models with iterative retrieval-generation synergy. In: Bouamor, H., Pino, J., Bali, K. (eds.) Findings of the Association for Computational Linguistics: EMNLP 2023, pp. 9248–9274. Association for Computational Linguistics, Singapore (2023). https://doi.org/10.18653/v1/2023.findings-emnlp.620, https://aclanthology.org/2023.findings-emnlp.620
22. Siriwardhana, S., Weerasekera, R., Wen, E., Kaluarachchi, T., Rana, R., Nanayakkara, S.: Improving the domain adaptation of retrieval augmented generation (RAG) models for open domain question answering. Trans. Assoc. Comput. Linguist. **11**, 1–17 (2023). https://doi.org/10.1162/tacl_a_00530, https://aclanthology.org/2023.tacl-1.1
23. Wang, B., et al.: Decodingtrust: A comprehensive assessment of trustworthiness in GPT models. In: Advances in Neural Information Processing Systems (NeurIPS) (2023)
24. Wang, C., Liu, S., Yang, H., Guo, J., Wu, Y., Liu, J.: Ethical considerations of using CHATGPT in health care. J. Med. Internet Res. **25**, e48009 (2023)
25. Williams, R., et al.: From transparency to accountability of intelligent systems: moving beyond aspirations. Data Policy **4**, e7 (2022). https://doi.org/10.1017/dap.2021.37
26. Winter, P.D., Carusi, A.: (De) troubling transparency: artificial intelligence (AI) for clinical applications. Med. Humanit. **49**(1), 17–26 (2023)
27. Xu, Y., Namazifar, M., Hazarika, D., Padmakumar, A., Liu, Y., Hakkani-Tur, D.: KILM: Knowledge injection into encoder-decoder language models. In: Rogers, A., Boyd-Graber, J., Okazaki, N. (eds.) Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 5013–5035. Association for Computational Linguistics, Toronto, Canada (2023). https://doi.org/10.18653/v1/2023.acl-long.275, https://aclanthology.org/2023.acl-long.275
28. Yu, Z., Xiong, C., Yu, S., Liu, Z.: Augmentation-adapted retriever improves generalization of language models as generic plug-in. In: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics, Volume 1: Long Papers, pp. 2421–2436. Association for Computational Linguistics (July 9–14 2023)
29. Zhang, Z., et al.: Iag: Induction-augmented generation framework for answering reasoning questions. In: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing. pp. 1–14, Association for Computational Linguistics (Dec 6–10 2023)

Beyond Self-consistency: Ensemble Reasoning Boosts Consistency and Accuracy of LLMs in Cancer Staging

Chia-Hsuan Chang[1], Mary M. Lucas[1], Yeawon Lee[1],
Christopher C. Yang[1](✉), and Grace Lu-Yao[2]

[1] College of Computing and Informatics, Drexel University, Philadelphia, PA, USA
{cc3859,mml367,yl3427,chris.yang}@drexel.edu

[2] Department of Medical Oncology, Sidney Kimmel Cancer Center, Thomas Jefferson University, Philadelphia, PA, USA
Grace.LuYao@jefferson.edu

Abstract. Pathologic cancer stage, crucial for treatment decisions, is often buried in unstructured pathology reports. This study investigates using pre-trained clinical LLMs for stage extraction, leveraging prompting techniques like chain-of-thought to enhance model transparency. While self-consistency methods further improve LLM performance, they can introduce inconsistencies in reasoning paths and predictions. We propose an ensemble reasoning approach, aiming for reliable cancer stage extraction. Utilizing an open-source clinical LLM on real-world reports, we demonstrate that the ensemble approach improves consistency and boosts performance, paving the way for utilizing LLMs in healthcare settings where reliability and interpretability are paramount.

Keywords: Large language model · Prediction consistency · Cancer stage classification · Pathology report

1 Introduction

The standardized TNM cancer staging system is an essential part of cancer diagnosis and management, classifying the extent of cancer based on tumor size (T), lymph node involvement (N), and distant metastasis (M). Pathology reports detail this information based on tissue sample analyses to determine the pathologic TNM (pTNM) stage, but their free-text format complicates large-scale, rapid data extraction. In our previous study [1], we found that large language models (LLMs) perform better than fine-tuned models without needing training samples which can be costly and time-consuming to prepare. LLMs can benefit from different prompt engineering techniques, like the Chain-of-Thought approach [4,7], which can improve the interpretability of the model's predictions. However, LLMs' stochastic generation process can lead to varying reasoning paths and inconsistent responses, which can be problematic in clinical scenarios where accountability is essential [5]. To address inconsistent responses, we develop an Ensemble-Reasoning (EnsReas) approach, building upon the concept of self-consistency [6]. EnsReas enhances LLM analysis of pathology reports

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 224–228, 2024.
https://doi.org/10.1007/978-3-031-66538-7_23

by first leveraging self-consistency as an intermediate step to generate multiple reasoning and prediction responses. These reasonings are then grouped by predicted outcomes, forming a revised prompt that guides the LLM to re-assess inconsistent initial answers, ensuring a more robust and reliable output. Our results demonstrate that EnsReas performs better than the baseline zero shot (ZS) and zero shot chain-of-thought (ZS-CoT) methods regarding predictive performance. Moreover, it outperforms ZS-CoT with self-consistency (ZS-CoT-SC) in predictive performance and consistency, suggesting that LLMs can enhance their decision-making process using the EnsReas approach, leading to more consistent and reliable responses for complex tasks like determining cancer staging from pathology reports. This advancement underscores LLMs' potential in interpreting and utilizing clinical data for effective cancer treatment planning.

2 Materials and Methods

Data and Language Model. We use a real-world corpus of breast cancer pathology reports from the Cancer Genomic Atlas (TCGA) project. The raw reports, in PDF format, are available from the National Cancer Institute (NCI) Genomic Data Commons (GDC) portal. We utilize a preprocessed subset of reports curated and described in [3] and made available for download[1]. We focus our experiments on predicting pathologic T and N stage for breast cancer because it is one of the top diagnosed cancers in the United States[2] and has good representation of reports and ground truths in the dataset. Because some of the pathology reports do not report all pTNM stages within the same report, we follow [2] and treat the T stage and N stage prediction as two different tasks. As a result, we have 1,031 reports for T category (T1:589, T2:273, T3:131, and T4:38) and 800 reports for N category (N0:316, N1:300, N2:110, and N3:74). We utilize Med42-70B[3], an open-access clinical LLM, installed on a local server. Med42-70B is derived from Llama2-70B and instruction-tuned on a dataset of medical knowledge, with reported superior performance in the zero shot setting compared to GPT-3.5. When compared with other open access clinical generative LLMs, Med42-70B outperforms ClinicalCamel-70B.

Baselines and Ensemble-Reasoning (EnsReas). We first implement three prompting strategies, zero shot (ZS), zero shot chain-of-thought (ZS-CoT) [4], and ZS-CoT with self-consistency (ZS-CoT-SC) [6], to obtain baseline performance for our task. For ZS prompting we provide the LLM with the report and instruction to return the predicted stage p for the pathologic T stage or the pathologic N stage. In ZS-CoT, we provide the report and instruct the LLM to first "think step by step" to retrieve the generated reasoning c, and treat c as the context for the LLM to predict the stage p. We adopt greedy decoding for ZS and ZS-CoT to get the most likely stage prediction p for each report, and we measure the performance of ZS and ZS-CoT based on their generated p for each

[1] https://github.com/tatonetti-lab/tcga-path-reports.

[2] https://seer.cancer.gov/statfacts/html/common.html.

[3] https://huggingface.co/m42-health/med42-70b.

report. As for ZS-CoT-SC, for each report we adopt temperature sampling[4] on ZS-CoT to obtain 10 responses from the LLM, denoted as (C, P), where C is a list of 10 generated reasonings and P is a list of 10 generated predicted stages. The majority vote (the most frequent answer) from P is considered to be the final prediction for a report, and is used to measure performance.

EnsReas requires the outputs of ZS-CoT-SC for each report r. Therefore, each report $r \in R$ has a list of 10 reasonings C and a list of 10 predicted answers P. By analyzing the P for each report, all reports can be automatically separated into reports with consistent predictions R^{con} and reports with inconsistent predictions R^{inc}. The R^{con} are reports that have only one unique predicted stage in their P, and the R^{inc} are a subset of reports by filtering out R^{con} from all reports. Because the predictions for reports in R^{con} are deterministic, the EnsReas keeps using the same predictions as ZS-CoT-SC.

For the reports in R^{inc}, we design a prompt for EnsReas to simulate a panel discussion, triggering the LLM to resolve the inconsistent reasonings and predictions of a given report. Specifically, for each report with various answers, we aggregate the reasonings by each answer, yielding a set of grouped reasonings: $g_p = \{c | c \text{ is reasoning for answer } p, c \in C, p \in P\}$, where g_p can be considered as a set of reasonings (opinions) from experts who choose p as the answer. Take a report with four different predicted stages (i.e., T1, T2, T3, and T4) as example, the following prompt demonstrates how we integrate g_p in defining the prompt:

prompt = """
Report: {report}

Panel Responses:
T1: {g_{T1}: reasonings that support T1 as the answer}
T2: {g_{T2}: reasonings that support T2 as the answer}
T3: {g_{T3}: reasonings that support T3 as the answer}
T4: {g_{T4}: reasonings that support T4 as the answer}

You are provided with the pathology report and the chosen answers from the panel of experts with the corresponding reasonings. The reasonings provided by the experts are aggregated by chosen answer.

Please review each report. Analyze the reasonings provided by the panel for the chosen answers. Keep in mind that the majority vote may not be the correct one, therefore you should review report carefully in addition to considering the panel reasonings.

The correct answer is
"""

[4] We set temperature as 0.7 and top p as 0.95 in our study.

With this prompt template, we instruct the LLM to review every report in R^{inc} and its grouped reasonings. To fairly compare with ZS-CoT-SC, EnsReas also adopts temperature sampling to have a set of 10 updated predictions for a report, denoted P^{update}. The most frequent answer in P^{update} will be used for performance evaluation. Moreover, to understand how EnsReas improves those inconsistent predictions made by ZS-CoT-SC, we will compare every report's P and P^{update} generated by ZS-CoT-SC and EnsReas, respectively.

Evaluation Metrics. To measure the predictive performance of each prompting strategy, we report macro precision, macro recall, and macro F1 score, where macro average is taken for considering the performances of all possible stages K in each category. We use entropy to measure the consistency of predictions generated by ZS-CoT-SC and EnsReas. Specifically, given a report r, we have 10 predictions (P) generated by ZS-CoT-SC and 10 predictions ($P^{updated}$) generated by EnsReas. We then determine the consistency by averaging the entropy of all reports: $\frac{\sum_{r \in R} \text{entropy}(p_1^r, \ldots, p_{10}^r)}{R}$, where $p_i^r \in P, P^{update}$ and $i \in [1, 10]$. A higher average entropy suggests a method has higher inconsistency in the predictions.

3 Result and Discussion

Table 1 presents the performance comparison of our proposed EnsReas and the other baselines. **In terms of predictive performance**, ZS has the worst performance among all strategies. ZS-CoT has a significantly increased F1 in T category while its F1 is comparable with ZS. When comparing between ZS-CoT and ZS-CoT-SC, ZS-CoT-SC only has slightly higher and comparable macro F1 with ZS-CoT in T category and N category, respectively. EnsReas approach performs the best with highest F1 in both T and N categories. These results suggest that the large language model (i.e., Med42-70B in this study) is capable of refining its generations for delivering more accurate answers in the cancer staging task. **In terms of prediction consistency**, EnsReas generates more consistent predictions than ZS-CoT-SC[5], supported by EnsReas's significant lower average entropy value.

Table 1. Performance of each prompting strategy. All precision, recall, f1-score are macro average across different classes.

	T Category					N Category				
	Precision (↑)	Recall (↑)	F1 (↑)	Consistency (↓)	Support	Precision (↑)	Recall (↑)	F1 (↑)	Consistency (↓)	Support
ZS	0.725	0.725	0.688		1031	0.848	**0.810**	0.823		800
ZS-CoT	0.855	0.740	0.790		1031	0.873	0.788	0.825		800
ZS-CoT-SC	**0.865**	0.738	0.793	0.162	1031	0.868	0.795	0.825	0.093	800
EnsReas	0.860	**0.755**	**0.800**	**0.036**	1031	**0.875**	0.808	**0.838**	**0.023**	800

[5] Since EnsReas depends on ZS-CoT-SC to generate a set of reasonings as input, we consider the predictive outcomes of ZS-CoT-SC as the reference to demonstrate the changes of consistency achieved by EnsReas.

4 Conclusion and Future Work

In this work, we proposed and investigated the use of EnsReas to improve the consistency and performance of LLMs applied to a clinical task: determining cancer stage from pathology reports. Our experimental results indicate that EnsReas not only generates more accurate predictions for cancer staging but also reduces inconsistencies in LLM outputs, addressing a significant concern regarding the reliability of LLM-based predictions in clinical settings. Future research is needed to explore the application of EnsReas to a broader range of clinical tasks, perform qualitative analysis on EnsReas generated reasonings, and investigate mechanisms to incorporate clinician feedback into EnsReas.

Acknowledgement. This work was supported in part by the National Science Foundation under the Grants IIS-1741306 and IIS-2235548, and by the Department of Defense under the Grant DoD W91XWH-05-1-023. This material is based upon work supported by (while serving at) the National Science Foundation. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

1. Chang, C.H., Lucas, M.M., Lu-Yao, G., Yang, C.C.: Classifying cancer stage with open-source clinical large language models. arXiv (2024). https://doi.org/10.48550/arXiv.2404.01589
2. Kefeli, J., Tatonetti, N.: Generalizable and automated classification of TNM stage from pathology reports with external validation. medRxiv (2023). https://doi.org/10.1101/2023.06.26.23291912
3. Kefeli, J., Tatonetti, N.: TCGA-Reports: a machine-readable pathology report resource for benchmarking text-based AI models. Patterns (2024)
4. Kojima, T., Gu, S.S., Reid, M., Matsuo, Y., Iwasawa, Y.: Large language models are zero-shot reasoners. arXiv (2023). https://doi.org/10.48550/arXiv.2205.11916
5. Sivarajkumar, S., Kelley, M., Samolyk-Mazzanti, A., Visweswaran, S., Wang, Y.: An empirical evaluation of prompting strategies for large language models in zero-shot clinical natural language processing. arXiv (2023). https://doi.org/10.48550/arXiv.2309.08008
6. Wang, X., et al.: Self-consistency improves chain of thought reasoning in language models. In: The Eleventh International Conference on Learning Representations, ICLR (2023)
7. Wei, J., et al.: Chain-of-thought prompting elicits reasoning in large language models. arXiv (2023). https://doi.org/10.48550/arXiv.2201.11903

Clinical Reasoning over Tabular Data and Text with Bayesian Networks

Paloma Rabaey[1](✉), Johannes Deleu[1], Stefan Heytens[2], and Thomas Demeester[1]

[1] IDLab, Department of Information Technology, Ghent University - imec, Ghent, Belgium
paloma.rabaey@ugent.be

[2] Department of Public Health and Primary Care, Ghent University, Ghent, Belgium

Abstract. Bayesian networks are well-suited for clinical reasoning on tabular data, but are less compatible with natural language data, for which neural networks provide a successful framework. This paper compares and discusses strategies to augment Bayesian networks with neural text representations, both in a generative and discriminative manner. This is illustrated with simulation results for a primary care use case (diagnosis of pneumonia) and discussed in a broader clinical context.

Keywords: Clinical reasoning · Bayesian networks · Neural networks · Text representations

1 Introduction

The process of clinical reasoning lies at the heart of many interactions between a clinician and their patient [22]. Clinical reasoning is the process by which a clinician integrates their own knowledge with patient information (like symptoms, objective medical evidence, background, medical history...), to arrive at a diagnosis and subsequent therapeutic options [6]. Cognitive biases and knowledge deficits can cause errors in clinical reasoning, causing the clinician to arrive at an incorrect diagnosis [13]. To help clinicians avoid these pitfalls, it can help to (partially) automate the process of clinical reasoning [2,21]. Bayesian networks (BNs) are ideally suited for this task, given (i) their ability to model complex problems involving uncertainty, (ii) their ability to combine data and expert knowledge, and (iii) their interpretable graphical structure [8]. However, a key factor limiting the adoption of BNs in clinical practice is their inadequacy to deal with realistic medical data [7], often a mix between structured tabular variables (disease codes, timestamps, demographic features, lab results...) and unstructured text (consultation notes, discharge summaries...) [5]. Encoding the information contained in the unstructured text into structured variables not only requires (considerable) effort, but also inevitably results in loss of information.

In this work, we explore how to integrate unstructured text data in Bayesian networks, to facilitate joint clinical reasoning over structured tabular data and

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 229–250, 2024.
https://doi.org/10.1007/978-3-031-66538-7_24

unstructured text. To this end, we investigate a relevant use case in primary care: diagnosis of pneumonia. We create an artificial yet realistic dataset, allowing us to control several aspects of the data generation process. This allows us to investigate the impact of different modeling approaches to integrate text in the clinical reasoning process, and discuss their advantages and pitfalls. By keeping the use case highly tangible for a clinical audience, we aim to lower the bar toward real-world medical applications of the presented technology.

Our main contribution is the study of different approaches to integrate the neural representation of a textual variable in the BN. In particular, we compare the properties of adding the text with a generative model (in the space of neural text representations, fitted alongside the BN) vs. a discriminative model (a text classifier jointly trained with the BN). We evaluate the performance of both approaches on the prediction of pneumonia in a toy setting, and compare with baselines which are either missing the text component or the BN structure. Based on the presented results, we discuss (i) the advantages of including unstructured text, (ii) the properties of different approaches to achieve this, and (iii) the overall idea of performing Bayesian inference for automated clinical reasoning involving textual data.

2 Related Work

Since the topic of this paper touches on multiple different research domains, we position our work in regards to the most relevant domains, without providing an exhaustive overview of all related research.

Clinical Reasoning: This work follows the interpretation of clinical reasoning as an analytical process, where a clinician weighs up every piece of evidence to reject or confirm a certain diagnostic hypothesis [20,23]. Starting from a set of differential diagnoses, each with their own prior probability reflecting their prevalence in the population, clinical reasoning comes down to updating the likelihood of each diagnosis with every new piece of evidence that comes in, using Bayes' rule. This results in a posterior likelihood for each diagnosis, which the clinician takes into account for planning further steps.

Bayesian Networks: BNs form the perfect tool to formalize the process outlined above [8]. Their interpretable graph structure can help keep track of independencies between certain types of evidence and particular diagnoses, and inference in BNs follows Bayes' rule. BNs have been used to model a wide range of medical conditions in research settings [11], including respiratory diseases such as pneumonia and Covid-19 [4]. However, their deployment for clinical decision making in practice remains limited, partly due to real-world data challenges [7].

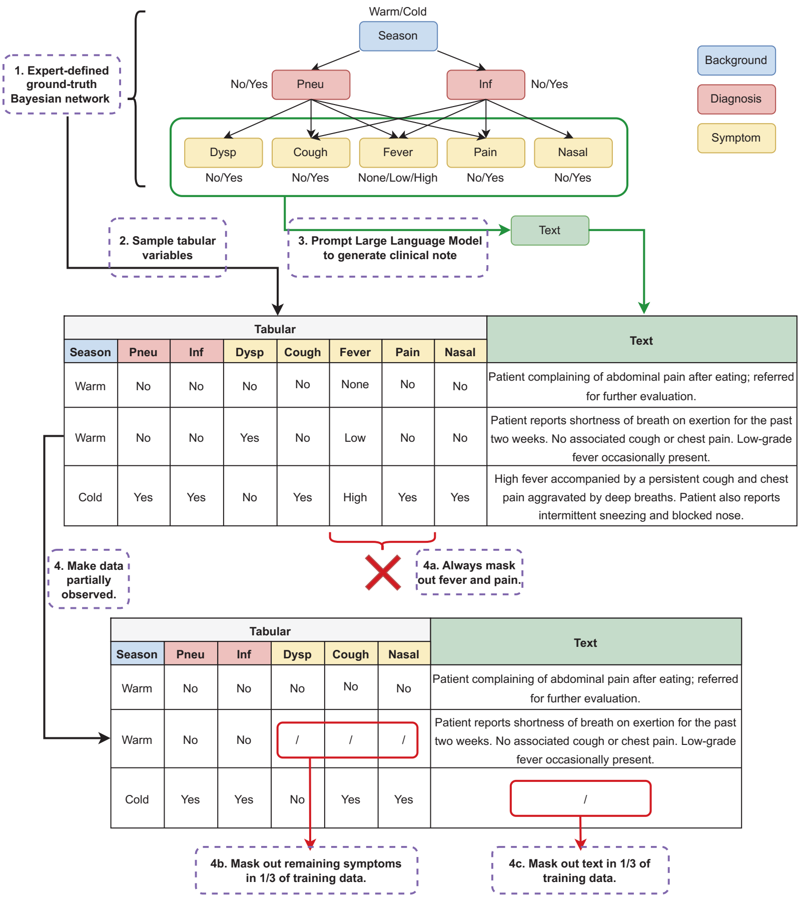
Clinical Unstructured Text: The last few decades have seen an abundance of electronic medical records being collected in clinical practice, which form a useful source of data to build clinical decision support systems (CDSS). These records are usually made up of structured data (disease codes, dates, treatment codes...), as well as free text [5]. Studies have shown that ignoring the information present in free text records can results in data loss and bias in CDSS [16]. Nevertheless, a large majority of CDSS either completely disregards this unstructured text [15] or applies information extraction techniques to turn the text into tabular format, which then serves as input to the CDSS [5]. Turning unstructured text into structured variables using information extraction methods (see, e.g., [19]) and then building a predictive model on top of the structured features has been applied to learning clinical BNs in the past [18,24]. Our work focuses on integrating the full unstructured text, removing the need for this information extraction step. Some CDSS are built on raw unstructured text, yet they often fail to integrate it with the structured portion of the data [12,15]. Zhang et al. manage to successfully integrate both through a multi-modal recurrent neural network that combines embeddings of clinical text with static and time-varying features in the electronic medical record, outputting a full patient representation that can be used for further downstream prediction [25]. We also represent clinical text through neural representations, though we use a BN in combination with feedforward neural components to integrate these text embeddings with the static tabular features in the reasoning process.

Neuro-Symbolic AI: The integration of reasoning and learning has seen considerable progress in recent years, in the field of Neuro-Symbolic (NeSy) AI [10]. One strongly related contribution is the DeepProbLog (DPL) framework [9]. The authors show how a probabilistic logic program can be extended with neural predicates, whereby a neural network converts an unstructured data item (like an image) into class probabilities, that are then treated as regular predicates in the logic program. Importantly, the parameters of the logic program and of the neural networks encoding unstructured data can be jointly trained. The discriminative model for integrating text nodes into BNs in this paper corresponds to the neural predicates approach, since a BN can be seen as a special case of a probabilistic program. In contrast, we also compare this approach with a generative model in text representation space, and provide a targeted discussion from the perspective of clinical reasoning.

3 Use Case and Data Description

Our use case focuses on diagnostic clinical reasoning performed by a general practitioner (GP) [23]. One non-trivial task in primary care is distinguishing pneumonia from an upper respiratory tract infection (also known as the common cold), where the former is more serious and calls for treatment with antibiotics. When a patient presents with respiratory symptoms, a GP will apply clinical reasoning based on these symptoms and a short clinical examination, ordering the necessary additional testing or starting a treatment only if the probability for pneumonia exceeds a certain threshold.

We create an artificial dataset that allows us to study automation of the clinical reasoning process for the pneumonia use case, in the presence of unstructured text resembling consultation notes taken by the GP during a patient encounter. Figure 1 shows the data generation process. Its caption describes the four key


1. Expert-defined ground-truth Bayesian network
Warm/Cold
Season
No/Yes
Pneu
Inf
No/Yes
Dysp
Cough
Fever
Pain
Nasal
No/Yes
No/Yes
None/Low/High
No/Yes
No/Yes
Background
Diagnosis
Symptom
2. Sample tabular variables
3. Prompt Large Language Model to generate clinical note
Text
4. Make data partially observed.
4a. Always mask out fever and pain.
4b. Mask out remaining symptoms in 1/3 of training data.
4c. Mask out text in 1/3 of training data.

Tabular								Text
Season	Pneu	Inf	Dysp	Cough	Fever	Pain	Nasal	
Warm	No	No	No	No	None	No	No	Patient complaining of abdominal pain after eating; referred for further evaluation.
Warm	No	No	Yes	No	Low	No	No	Patient reports shortness of breath on exertion for the past two weeks. No associated cough or chest pain. Low-grade fever occasionally present.
Cold	Yes	Yes	No	Yes	High	Yes	Yes	High fever accompanied by a persistent cough and chest pain aggravated by deep breaths. Patient also reports intermittent sneezing and blocked nose.

Tabular						Text
Season	Pneu	Inf	Dysp	Cough	Nasal	
Warm	No	No	No	No	No	Patient complaining of abdominal pain after eating; referred for further evaluation.
Warm	No	No	/	/	/	Patient reports shortness of breath on exertion for the past two weeks. No associated cough or chest pain. Low-grade fever occasionally present.
Cold	Yes	Yes	No	Yes	Yes	/

Fig. 1. Key steps in generating the artificial dataset, where each sample consists of both tabular variables and corresponding clinical text descriptions. With help of an expert, we define a Bayesian network (BN) simulating the pneumonia use case (**step 1**). We sample the tabular variables (background, diagnoses and symptoms) from the distribution defined by this BN (**step 2**), prompting a large language model (GPT3.5 [14]) to generate realistic but fictitious consultation notes based on the sampled symptoms (**step 3**). We repeat steps 1 to 3 to generate 4000 training samples and 1000 test samples. Finally, to induce property (ii) of realistic medical data (see Sect. 3), we remove two symptoms, *fever* and *pain*, from the tabular portion of the data, ensuring they are never encoded and only observed through the text (**step 4a**). From now on, when we talk about symptoms, we take this to mean the symptoms *dysp*, *cough*, *nasal*, unless explicitly stated otherwise. For the training set only, we partially mask out the remaining symptoms (**step 4b**) and the text (**step 4c**) in a complementary subset of the training samples. Each sample now represents a fictional patient encounter, consisting of one background feature (*season*), two diagnoses (*pneu* and *inf*), three symptoms (*dysp*, *cough* and *nasal*, partially unobserved) and a textual description (*text*, partially unobserved). The text contains additional context on the three encoded symptoms, as well as describing two additional unencoded symptoms *fever* and *pain*.

steps, and a more detailed explanation is given in Appendices A and B. We aim to mirror the following properties of realistic medical data: (i) the data contains structured tabular variables and/or unstructured text, (ii) information in the text is only partially encoded in the structured variables, and (iii) the text contains additional context on the patient's background and symptoms, complementing the information encoded in the tabular variables. The final train and test datasets are available in our Github repository: https://github.com/prabaey/bn-text.

4 Augmenting BNs with Text Representations

We propose two model architectures that are able to integrate text in a Bayesian network: **`BN-gen-text`** (Sect. 4.1) and **`BN-discr-text`** (Sect. 4.2). Both models incorporate text through a single-vector text embedding, either modeling its distribution directly or learning classifiers with these representations as an input. As shown in Fig. 2, we compare them with three baseline models. Our first baseline **`BN`** is a standard Bayesian network without text variables, trained only on the partially observed tabular features. Its extension **`BN++`** is trained on a version of the training set where the symptoms $fever$ and *pain* are exceptionally *not* masked out, forming an upper bound to the performance of all other models, which never get to directly observe these two symptoms. The last baseline **`FF-discr-text`** is a discriminative feed-forward neural network which takes both tabular features (one-hot encoded) and text (as a *BioLORD* embedding) as an input, and outputs a prediction for *pneu* or inf. Details on the baseline models **`BN`**, **`BN++`** and **`FF-discr-text`** can be found in Appendix C.1 and C.2.

All models are trained on the final dataset shown in Fig. 1 (with only the symptoms *dysp*, *cough* and *nasal*, partially observed) except for **`BN++`** (where $fever$ and *pain* are added, as described above). During inference, each model computes a posterior distribution for each diagnosis given some set of evidence. For readability, we represent the diagnoses by D_i ($i \in \{0, 1\}$) with D_0 (*pneu*) and D_1 (inf), symptoms as S_0 (*dysp*), S_1 (*cough*) and S_2 (*nasal*), background as B (*season*) and text as T. We discuss how each model is able to calculate the following posterior diagnostic probabilities:

- $\mathcal{P}(D_i \mid B, S_0, S_1, S_2)$: take only background and symptoms as evidence.
- $\mathcal{P}(D_i \mid B, S_0, S_1, S_2, T)$: take background, symptoms and text as evidence.
- $\mathcal{P}(D_i \mid B, T)$: take background and text as evidence.

4.1 Bayesian Network with Text Generator (`BN-gen-text`)

Training: In the **`BN-gen-text`** model, a text node is added to the **`BN`** baseline, conditioned on all diagnoses and symptoms. The conditional distributions for all tabular variables are trained using Maximum Likelihood Estimation, as a standard Bayesian network (see Appendix C.1). To obtain a vector representation for the text, we use *BioLORD*, which is a pre-trained language model that produces semantic single-vector representations for clinical sentences and biomedical concepts [17]. 32 separate multivariate Gaussians, one for each possible combination of the values for the two diagnoses and three symptoms, are fitted to the text embeddings to obtain the distribution $\mathcal{P}(T \mid D_0, D_1, S_0, S_1, S_2)$. This basic model allows us to get the probability density of unseen text embeddings and even sample new ones, although those cannot be directly decoded into text. To learn each Gaussian, we select all samples in the training set that match a particular condition and fit the mean μ and covariance matrix $\boldsymbol{\Sigma}$ to the corresponding text embeddings. The estimated covariance matrix Σ is regularized as follows

$$\mathcal{P}(T \mid D_0, D_1, S_0, S_1, S_2) = \mathcal{N}_{D_0,D_1,S_0,S_1,S_2}(\boldsymbol{\mu}, (1-\alpha)\boldsymbol{\Sigma} + \alpha \boldsymbol{I}) \qquad (1)$$

where the hyperparameter α allows tuning the contribution of the individual variances of the text representation dimensions.

Inference: After training, we can calculate the posterior for either diagnosis D_i given a set of evidence by applying Bayes' rule and marginalizing over the learned joint distribution. For $\mathcal{P}(D_i \mid B, S_0, S_1, S_2)$ the calculation is the same as in a standard BN, since the conditional text distribution $\mathcal{P}(T \mid D_0, D_1, S_0, S_1, S_2)$ is integrated out. $\mathcal{P}(D_i \mid B, S_0, S_1, S_2, T)$ and $\mathcal{P}(D_i \mid B, T)$ on the other hand do evaluate the conditional text distribution. The detailed equations for all posteriors can be found in Appendix C.3.

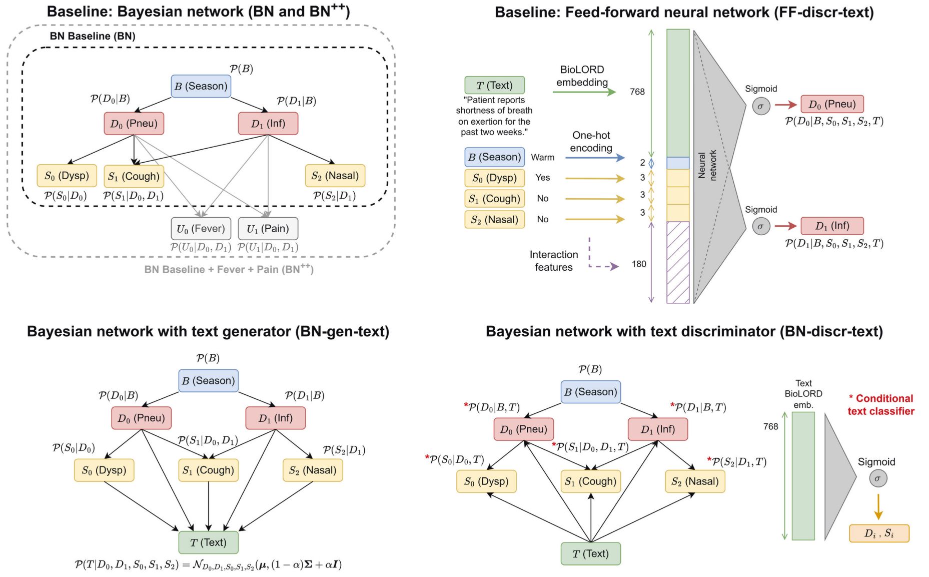

Baseline: Bayesian network (BN and BN++)
BN Baseline (BN)
$\mathcal{P}(B)$
B (Season)
$\mathcal{P}(D_0|B)$
$\mathcal{P}(D_1|B)$
D_0 (Pneu)
D_1 (Inf)
S_0 (Dysp)
S_1 (Cough)
S_2 (Nasal)
$\mathcal{P}(S_0|D_0)$
$\mathcal{P}(S_1|D_0, D_1)$
$\mathcal{P}(S_2|D_1)$
U_0 (Fever)
U_1 (Pain)
$\mathcal{P}(U_0|D_0, D_1)$
$\mathcal{P}(U_1|D_0, D_1)$
BN Baseline + Fever + Pain (BN++)
Baseline: Feed-forward neural network (FF-discr-text)
T (Text)
"Patient reports shortness of breath on exertion for the past two weeks."
BioLORD embedding
768
One-hot encoding
B (Season)
Warm
S_0 (Dysp)
Yes
S_1 (Cough)
No
S_2 (Nasal)
No
2
3
3
3
Interaction features
180
Neural network
Sigmoid
σ
D_0 (Pneu)
$\mathcal{P}(D_0|B, S_0, S_1, S_2, T)$
Sigmoid
σ
D_1 (Inf)
$\mathcal{P}(D_1|B, S_0, S_1, S_2, T)$
Bayesian network with text generator (BN-gen-text)
$\mathcal{P}(B)$
B (Season)
$\mathcal{P}(D_0|B)$
$\mathcal{P}(D_1|B)$
D_0 (Pneu)
D_1 (Inf)
$\mathcal{P}(S_0|D_0)$
$\mathcal{P}(S_1|D_0, D_1)$
$\mathcal{P}(S_2|D_1)$
S_0 (Dysp)
S_1 (Cough)
S_2 (Nasal)
T (Text)
$\mathcal{P}(T|D_0, D_1, S_0, S_1, S_2) = \mathcal{N}_{D_0,D_1,S_0,S_1,S_2}(\boldsymbol{\mu}, (1-\alpha)\boldsymbol{\Sigma} + \alpha \boldsymbol{I})$
Bayesian network with text discriminator (BN-discr-text)
$\mathcal{P}(B)$
B (Season)
$^*\mathcal{P}(D_0|B, T)$
$^*\mathcal{P}(D_1|B, T)$
D_0 (Pneu)
D_1 (Inf)
$^*\mathcal{P}(S_1|D_0, D_1, T)$
$^*\mathcal{P}(S_0|D_0, T)$
$^*\mathcal{P}(S_2|D_1, T)$
S_0 (Dysp)
S_1 (Cough)
S_2 (Nasal)
T (Text)
Text BioLORD emb.
768
* Conditional text classifier
Sigmoid
σ
D_i , S_i

Fig. 2. Schematic depiction of all models. The top row presents our baselines `BN`, `BN++` and `FF-discr-text`. The bottom row shows `BN-gen-text` and `BN-discr-text`, two variants of a Bayesian network augmented with text representations.

4.2 Bayesian Network with Text Discriminator (`BN-discr-text`)

Training: In the **`BN-discr-text`** model, we augment the **`BN`** baseline by conditioning all diagnoses and symptoms on the text embedding. This contrasts with the **`BN-gen-text`** approach, where we augment the **`BN`** baseline with distributions of text embeddings conditioned on the diagnoses and symptoms. While this arc reversal renders the two BNs semantically non-equivalent, all independence relations between the non-text nodes remain intact. Each of the conditional distributions is modeled as a set of discriminative neural text classifiers, one for each configuration of the tabular parent variables, meaning there are 12 in total. For example, we model $\mathcal{P}(D_0 \mid B = \textit{warm}, T)$ and $\mathcal{P}(D_0 \mid B = \textit{cold}, T)$ as two separate feed-forward neural networks that take the *BioLORD* text embedding as an input, and learn to predict the diagnosis probability for D_0 at the output. All parameters are optimized jointly by maximizing the likelihood $\mathcal{P}(B, D_0, D_1, S_0, S_1, S_2 \mid T)$ (see Appendix C.4) based on the training data. By making this likelihood conditional on text, we refrain from having to learn a prior distribution $\mathcal{P}(T)$ of the text embeddings.

Inference: $\mathcal{P}(D_i \mid B, S_0, S_1, S_2, T)$ is again obtained by applying Bayes' rule and marginalizing over the joint distribution (conditional on text). The trained classifiers are used to evaluate the probabilities needed during inference. Strictly speaking, conditioning on the text node means that $\mathcal{P}(D_i \mid B, S_0, S_1, S_2)$ cannot be computed. We circumvent this issue by conditioning on the embedding of the empty text "" in case no text is observed. The classifiers learn to take this into account, since an empty text occurs in 1/3 of the training data. Finally, $\mathcal{P}(D_i \mid B, T)$ is simply the output of one of the two diagnosis classifiers. As before, the detailed equations for all posteriors can be found in Appendix C.4.

5 Empirical Results and Analysis

Evaluation: There are various ways to measure the models' ability of estimating diagnostic probabilities, given the observed background, symptoms and/or textual inputs. This section assumes the real-world scenario with a known (binary) diagnosis on the test set, but no knowledge of ground truth conditional probabilities. We therefore rank all patients in the test set according to the estimated probability of the considered diagnosis, and measure the area under the precision-recall curve for that ranking by comparing with the binary ground truth, i.e., we report the average precision. Results are averaged over 5 training runs with different model initializations. The full code is available in our Github repository: https://github.com/prabaey/bn-text.

Results: Table 1 presents average precision results for the diagnosis of pneumonia (D_0). Corresponding results for the prediction of upper respiratory tract infection (D_1) are shown in Table 2 in Appendix D. Ablation results in terms of connectivity in the network are provided and discussed in Appendix D.2, and details on training and hyperparameter optimization are given in Appendix D.1.

Analysis: Comparing $\mathcal{P}(D_0 \mid B, S_0, S_1, S_2, T)$ and $\mathcal{P}(D_0 \mid B, S_0, S_1, S_2)$ in Table 1, we note that both **`BN-gen-text`** and **`BN-discr-text`** improve over the baseline **`BN`**. This improvement is thanks to the incorporation of text, which contains information on the symptoms $fever$ and *pain* that is otherwise never explicitly encoded in the tabular features, yet very useful for diagnosing *pneu*. Indeed, when **`BN-discr-text`** takes both symptoms and text into account, in $\mathcal{P}(D_0 \mid B, S_0, S_1, S_2, T)$, its average precision comes close to the upper bound set by the baseline **`BN++`**, with direct access to all 5 encoded symptoms. Furthermore, the ablation study in Appendix D.2 shows a dramatic performance drop for both **`BN-discr-text`** and **`BN-gen-text`** when omitting the direct relation between diagnoses and text in the network, rendering the model unable to incorporate any complementary text-only information (i.e., on $fever$ and *pain*) during inference.

Though the **`BN-gen-text`** model performs better than the baseline **`BN`**, we see two reasons for why it is not on par with **`BN-discr-text`**. First, the distribution $\mathcal{P}(T \mid D_0, D_1, S_0, S_1, S_2)$ is made up of 32 conditional Gaussians, each trained on a different subset of text embeddings that occur with a particular (possibly rare) combination of symptom and diagnosis values. The **`BN-discr-text`** model has a more modular architecture and does not suffer as much from limited relevant training samples to fit each of its classifiers. Second, a multivariate normal distribution is not the best fit for the text embeddings. This is probably also why we see a bigger performance gap between the prediction that incorporates only text ($P(D_0 \mid B, T)$) and the one that incorporates both text and symptoms ($P(D_0 \mid B, S_0, S_1, S_2, T)$) for the **`BN-gen-text`** model.

Table 1. Average precision over test set of three posterior probabilities for the diagnosis D_0 (*pneu*), each taking a different set of evidence into account (various combinations of background, symptoms and text). We show mean (± std) over 5 initialization seeds.

Model	Average precision for *pneu*		
	$P(D_0 \mid B, S_0, S_1, S_2, T)$	$P(D_0 \mid B, S_0, S_1, S_2)$	$P(D_0 \mid B, T)$
`BN`	-	0.0914 (± 0.0000)	-
`BN++`	-	**0.8326** (± 0.0000)	-
`FF-discr-text`	0.6574 (± 0.0118)	0.1090 (± 0.0020)	0.6220 (± 0.0121)
`BN-gen-text`	0.5870 (± 0.0000)	0.0892 (± 0.0007)	0.4434 (± 0.0000)
`BN-discr-text`	**0.7538** (± 0.0323)	0.1079 (± 0.0011)	**0.6922** (± 0.0273)

Interestingly, the **`FF-discr-text`** baseline performs worse than **`BN-discr-text`**. While the two *pneu* classifiers in **`BN-discr-text`** can focus on modeling the text given one particular value of the background variable, the **`FF-discr-text`** classifier needs to deal with various configurations of the background and symptoms, missing or present, as well as their interactions with the text, in a single model. This is why **`BN-discr-text`** already improves over **`FF-discr-text`**, even when it only incorporates text during inference. When **`BN-discr-text`** incorporates symptoms on top of text ($\mathcal{P}(D_0 \mid B, S_0, S_1, S_2, T)$), it improves over **`FF-discr-text`** with almost 10% points in average precision, proving the merit of learning separate symptom classifiers, as well as diagnosis classifiers, and incorporating all their predictions through a Bayesian inference procedure.

Table 2 in Appendix D shows higher average precision measures for the prediction of inf, which is much more common in our dataset than *pneu*. Note that including the text in the prediction does not improve performance. Indeed, the small gap between **`BN`** and **`BN++`** shows that knowledge on the symptoms $fever$ and *pain* does not improve the prediction of inf much. Their textual representation is therefore expected to have little impact as well.

6 Discussion and Conclusion

We conclude with a discussion of how the results from the previous section can be seen in a broader context, by answering three key questions on the integration of text in Bayesian networks (BNs) for clinical reasoning.

What are different ways to integrate text into a BN, to allow for joint reasoning over unstructured text and structured tabular features?

We compared two architectures belonging to complementary model families: a BN with text generator (**`BN-gen-text`**) and a BN with text discriminator (**`BN-discr-text`**). An advantage of the **`BN-gen-text`** model is that it follows the causal structure of the text generation process, making it more intuitive to understand. However, to fit a generative model for the text embeddings, we need to make assumptions on the distribution which do not hold in practice. Conditional fitting of the Gaussians for every combination of diagnoses and symptoms also leads to a bad fit for rarer combinations. Both of these downsides translate to inferior performance of the generative model on our use case. However, alternative generative architectures are worth exploring in future research. The **`BN-discr-text`** model can benefit from the flexibility of the neural classifiers without requiring any assumptions on the distribution of the text embeddings. Its modular approach (with separate classifiers for the diagnoses and symptoms) allows for an intuitive integration of the observed symptoms into the Bayesian inference procedure.

What are the advantages of using unstructured text for clinical reasoning, compared to only using structured tabular features?

Reducing clinical text to a set of structured variables can be challenging, and inherently causes loss of information. By retaining the raw text and training the model to deal with this, the information extraction step is no longer required. This avoids the need to determine up front which variables are (1) relevant for any set of diagnoses, (2) mentioned frequently enough, and (3) can be extracted with sufficient accuracy. We simulated the presence of complementary information in the text with the symptoms $fever$ and *pain*. The models that included the text, were able to leverage information on those symptoms without explicitly including them as variables in the BN, which turned out especially beneficial for the rarer disease (pneumonia). This aligns with the intuition that specific symptoms related to rare diseases may not be encoded due to their infrequent presence, while at the same time being indispensable for accurate diagnosis.

What are the advantages of Bayesian inference for clinical reasoning, compared to approaches that don't contain a BN component?

BNs model each conditional distribution separately. This is not the case for the **`FF-discr-text`** baseline, which directly outputs a prediction for the diagnosis instead. This modular approach has multiple advantages. First of all, it helps the model deal with missing data during the training process: conditional distributions for variables that are not observed in a particular sample are simply not updated. The **`FF-discr-text`** baseline deals with unobserved symptoms by incorporating a special category in its one-hot encoding, which is much less natural. Second, and even more important, this modularity improves the interpretability of the prediction, which is essential in medical applications. An end user of the **`BN-discr-text`** model can inspect the outputs of the symptom classifiers as well as the diagnosis classifier, and see how all these probabilities contribute to the predicted posterior through the Bayesian inference process.

Acknowledgements. Paloma Rabaey's research is funded by the Research Foundation Flanders (FWO Vlaanderen) with grant number 1170122N. This research also received funding from the Flemish government under the "Onderzoeksprogramma Artificiële Intelligentie (AI) Vlaanderen" programme.

Appendix

A Data Generation Process

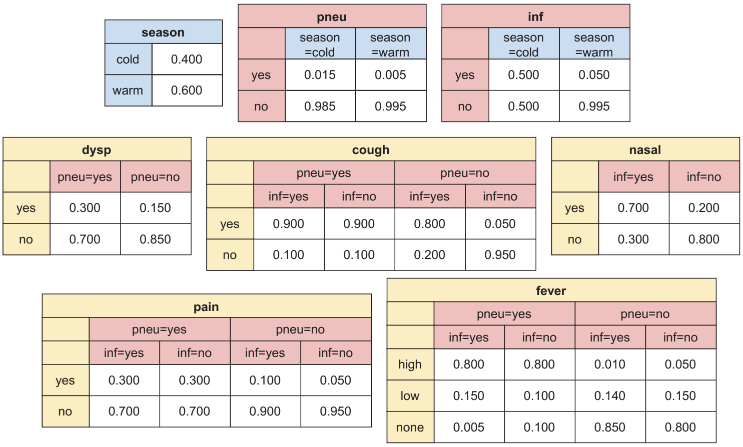
Figure 1 in the main text shows the steps we take to generate our dataset.

Step 1. Expert-Defined Bayesian Network. With help of an expert general practitioner, we define a Bayesian network (BN) that can be used for diagnosis of two diseases: pneumonia (*pneu*) and upper respiratory tract infection (inf). We model the effect of one background factor, *season* of the year, on both diagnoses. Additionally, five symptoms are added as nodes to the network: *dysp* (dyspnea, also known as shortness of breath), *cough*, $fever$, *pain* (chest pain and/or pain attributed to airways) and *nasal* (nasal congestion and/or sneezing). All variables are binary (*warm*/*cold* for *season* and *no*/*yes* for all others), except for $fever$, which can take on three levels (*none*/*low*/*high*). According to the expert, these five symptoms are the most informative to diagnose the two respiratory diseases in a primary care setting. Figure 3 shows the conditional probability tables (CPTs, defining the distribution of any child conditioned on its possible parent values), which were filled in according to the expert's own knowledge and experience. The product of all conditional distributions forms the joint ground truth distribution $\mathcal{P}_{GT}$ modeled by the BN, as shown in Eq. 2.

$$\begin{aligned}\mathcal{P}_{GT}(season, pneu, inf, dysp, cough, fever, pain, nasal) = \mathcal{P}_{GT}(season)\\ \mathcal{P}_{GT}(pneu \mid season)\mathcal{P}_{GT}(inf \mid season)\mathcal{P}_{GT}(dysp \mid pneu)\mathcal{P}_{GT}(cough \mid pneu, inf)\\ \mathcal{P}_{GT}(fever \mid pneu, inf)\mathcal{P}_{GT}(pain \mid pneu, inf)\mathcal{P}_{GT}(nasal \mid inf) \quad (2)\end{aligned}$$

Step 2. Sample Tabular Variables. We can easily sample from the joint distribution $\mathcal{P}_{GT}$ in a top-down fashion, starting from the root node *season*, generating the 2 diagnoses conditioned on *season*, and finally generating the 5 symptoms conditioned on the diagnoses. This way, we obtain 4000 train samples and 1000 test samples, each with 8 tabular features. We use the *pgmpy* library for implementing our BN and sampling from it [1].

Step 3. Generate Clinical Consultation Notes. We then prompt a large language model (LLM, in our case the GPT-3.5-turbo model from OpenAI [14]) to generate textual descriptions for each sample, given the presence or absence of the tabular symptoms. We want these textual descriptions to resemble clinical consultation notes made by a general practitioner for each fictitious patient encounter, which means the LLM only gets to observe the symptoms, not the diagnoses. Appendix B outlines the full prompting strategy.



season	
cold	0.400
warm	0.600

pneu	season =cold	season =warm
yes	0.015	0.005
no	0.985	0.995

inf	season =cold	season =warm
yes	0.500	0.050
no	0.500	0.995

dysp	pneu=yes	pneu=no
yes	0.300	0.150
no	0.700	0.850

cough	pneu=yes		pneu=no	
	inf=yes	inf=no	inf=yes	inf=no
yes	0.900	0.900	0.800	0.050
no	0.100	0.100	0.200	0.950

nasal	inf=yes	inf=no
yes	0.700	0.200
no	0.300	0.800

pain	pneu=yes		pneu=no	
	inf=yes	inf=no	inf=yes	inf=no
yes	0.300	0.300	0.100	0.050
no	0.700	0.700	0.900	0.950

fever	pneu=yes		pneu=no	
	inf=yes	inf=no	inf=yes	inf=no
high	0.800	0.800	0.010	0.050
low	0.150	0.100	0.140	0.150
none	0.005	0.100	0.850	0.800

Fig. 3. Conditional probability tables (CPTs) for all parent-child relations in the ground truth Bayesian network, as defined by an expert general practitioner.

Step 4. Make Data Partially Observed. While our dataset now fulfills desired properties (i) and (iii) as outlined in Sect. 3 in the main text, we still need to enforce property (ii), which we split up into 3 parts.

- **Step 4a:** Some symptoms are never encoded in the tabular portion of the data at all. To mimic this, we completely remove features $fever$ and *pain* from the dataset. This way, none of our models (except for baseline model **BN^{++}**, see later) ever observe these variables directly in tabular format, instead only seeing indirect mentions of them in the text. Both the symptoms $fever$ and *pain* are highly informative for the prediction of pneumonia, and any model that can extract information from the text should reap the benefits.
- **Step 4b:** Other symptoms are only encoded in a subset of the data. We simulate this situation by masking out the remaining symptoms in a subset of the training data. For 1/3 of the training data (1333 samples), we leave out the values for variables *dysp*, *cough* and *nasal*, rendering them unobserved. Note that we either observe all 3 symptoms or none at all, thereby avoiding the need to model missingness and simplifying reality.
- **Step 4c:** Furthermore, real data might not contain textual descriptions for all samples. For this reason, we mask out the textual description for another 1/3 of the training data (1333 samples). This leaves the remaining 1/3 of the training data (1334 samples), with fully observed symptoms and text.

We assume that the background variable is always observed – in a real setting, it could be extracted from the timestamp of the electronic record – and therefore never mask it out. The diagnoses variables are never masked out either, mostly to simplify the setup. We don't mask out anything in the test set, to leave full flexibility during the evaluation process in deciding what to include as input to the predictive models.

B Prompting Strategies

There are 5 symptoms in our tabular dataset, forming a total of 48 possible combinations. To mimic a realistic setting, we want each sample to have a unique textual description, meaning we need to generate a wide variety of different texts for each of these combinations of symptoms. We asked an expert general practitioner to provide us with some example patient encounter notes that could be used to prompt the LLM and encourage some variety in its responses. The expert received a symptom configuration and was asked to describe the patient encounter like they normally would in practice. We manually translated these notes from Dutch to English. We ensured that the top 10 most occurring symptom combinations in the training set (for which we will need to generate the highest number of unique textual descriptions) have at least one clinical example note. We had 20 annotated example notes in total. Since some of these notes are based on real encounters the general practitioner remembered from their own clinical practice, we do not make these public.

We now describe our LLM prompting strategy. We structure all prompts according to the OpenAI Chat Completions interface with the GPT-3.5-turbo model, using a temperature of 1 and a frequency penalty of 0.5. The full code to reproduce our prompting strategy is available in our Github repository: https://github.com/prabaey/bn-text.

Suppose we need to generate a text description for a symptom combination $\{dysp = d, cough = c, fever = f, pain = p, nasal = n\}$ that needs m unique textual descriptions. First, we check whether this combination is present in the set of examples. If one or more examples are found, we start our prompt with the requested symptom combination, followed by the example responses, see https://platform.openai.com/playground/p/poSdvoy9dipYIwVPXepRrUyL?model=gpt-3.5-turbo&mode=chat. If no example is found, we randomly pick two unrelated examples and prompt the language model by listing one after the other, preceded by their corresponding symptom combinations, see https://platform.openai.com/playground/p/6KOm6pP6DXmMwDxUJWdGGHP1?model=gpt-3.5-turbo&mode=chat. In both scenarios, after showing the examples, we ask the LLM to generate 5 clinical notes. We repeat the prompt as many times as needed to build up a set of m notes. To further encourage diversity in the responses, we only mention symptoms with positive values in the prompt in 50% of the cases, while in the other 50% we mention all symptoms and their values. We execute the entire pipeline separately for the train and test set. A random sample of the resulting notes were checked for coherence and correctness by the authors, which were deemed sufficient for this proof-of-concept setting.

We use a separate prompting strategy for the combination where all symptoms are absent. This combination occurs most often out of all, though we still only have 4 example notes for it. If we were to exclusively use the prompting strategy from scenario 1, the notes would have little variety. For this reason, we use 5 different strategies that each account for a different number of generated notes. While some strategies encourage the model to mention the absence of respiratory symptoms, others encourage the model to invent a completely unrelated patient encounter. We once again conduct the entire process for the train and test set separately, with the train set needing 1032 descriptions and the test set needing 388. We describe the 5 strategies we used, together with the proportion of textual descriptions we generated using each strategy and how many samples this comes down to in both the train and test set. These proportions were decided arbitrarily based on how useful we deemed each strategy to be.

1. Provide two in-context examples for symptom combination $\{dysp = no, cough = no, fever = none, pain = no, nasal = no\}$: https://platform.openai.com/playground/p/Zw9y4EZ8RRfGaZTgBCPu5DPT?model=gpt-3.5-turbo&mode=chat. (40%, Train: #400, Test: #151)
2. Provide two random out-of-context examples for other symptom combinations: https://platform.openai.com/playground/p/jYJeFfzXVZ5akndHGkB6cmPk?model=gpt-3.5-turbo&mode=chat (5%, Train: #50, #19
3. Similar to strategy 1, but do not mention the absent symptoms explicitly, thereby encouraging the model to describe cases outside of the respiratory domain: https://platform.openai.com/playground/p/Ikszv18Zbqr162kF0iSErbCT?model=gpt-3.5-turbo&mode=chat. We manually go over the generated cases and filter those out where *dysp*, *cough*, *fever*,

pain and *nasal* are described as present in the patient. (10%, Train: #107, Test: #39)

4. Same as strategy 3, but don't show any examples: https://platform.openai.com/playground/p/8KaVpr7chxLMHyeKz6Y2mMVE?model=gpt-3.5-turbo&mode=chat. We manually go over the generated cases and filter those out where *dysp*, *cough*, *fever*, *pain* and *nasal* are described as present in the patient. (10%, Train: #111, Test: #43)
5. Same as stragegy 4, still without showing any examples, but this time telling the model that the patient is not experiencing the symptoms *dysp*, *cough*, *fever*, *pain* and *nasal*: https://platform.openai.com/playground/p/eXDqpwMkqcvUA8wWiW1H06et?model=gpt-3.5-turbo&mode=chat. (35%, Train: #364, Test: #136)

C Augmenting BNs with Text Representations

C.1 Baseline: Bayesian Network (BN and BN^{++})

Training We train a simple Bayesian network (**BN**) where the Directed Acyclic Graph (DAG), which defines the structure between all the tabular variables, is the same as the one used to generate the data (see Fig. 1, excluding the unobserved symptoms). This Bayesian network defines the joint distribution as a product of six conditional distributions, one for each variable, as shown in Eq. 3. These distributions are learned from the training data using maximum likelihood estimation. This method studies the co-occurrence of particular values of each variable and its parents in the training set, filling up the CPTs as such. We use a K2 prior as a smoothing strategy, to counteract the extremely skewed probability distributions that might be learned when particular combinations of variables are never observed in the training set. We use the *pgmpy* Python library to learn the Bayesian network [1].

$$\mathcal{P}(B, D_0, D_1, S_0, S_1, S_2) = \mathcal{P}(B)\mathcal{P}(D_0 \mid B)\mathcal{P}(D_1 \mid B)\mathcal{P}(S_0 \mid D_0) \\ \mathcal{P}(S_1 \mid D_0, D_1)\mathcal{P}(S_2 \mid D_1) \quad (3)$$

Inference. The baseline Bayesian network can only include background and symptoms as evidence (no text). We can calculate the posterior for either diagnosis D_i by applying Bayes' rule and performing marginalization over the variables which are not included in the evidence, as shown in Eq. 4. We use the Variable Elimination method from *pgmpy* to perform exact inference.

$$\mathcal{P}(D_i \mid B, S_0, S_1, S_2) = \frac{\sum_{D_{1-i}} \mathcal{P}(B, D_0, D_1, S_0, S_1, S_2)}{\sum_{D_0, D_1} \mathcal{P}(B, D_0, D_1, S_0, S_1, S_2)} \quad (4)$$

Inclusion of Unobserved Symptoms. We also build a second variant of this baseline (**BN^{++}**) where we additionally include the unobserved symptoms *fever* and *pain* in the DAG. As opposed to all other models, this model is trained on a version of the training data where these two variables are *not* masked out. This baseline serves as an upper bound to the performance of all other models, which never get to directly observe these two symptoms. Equation 5 shows modeled the joint distribution, where U_0 and U_1 represent the unobserved symptoms $fever$ and *pain* respectively. Equation 4 can be trivially extended to obtain $\mathcal{P}(D_i \mid B, S_0, S_1, S_2, U_0, U_1)$, where evidence does not only include the partially observed symptoms S_i, but also the unobserved symptoms U_i.

$$\begin{aligned}\mathcal{P}(B, D_0, D_1, S_0, S_1, S_2, U_0, U_1) = \mathcal{P}(B)\mathcal{P}(D_0 \mid B)\mathcal{P}(D_1 \mid B)\mathcal{P}(S_0 \mid D_0)\\ \mathcal{P}(S_1 \mid D_0, D_1)\mathcal{P}(S_2 \mid D_1)\mathcal{P}(U_0 \mid D_0, D_1)\mathcal{P}(U_1 \mid D_0, D_1)\end{aligned} \tag{5}$$

C.2 Baseline: Feed-Forward Neural Network (`FF-discr-text`)

Training We train two discriminative feed-forward neural networks (`FF-discr-text`) which receive a vector representation of both the tabular features and the text at the input, and transform it into a one-dimensional representation which is turned into a prediction for $\mathcal{P}(D_i \mid B, S_0, S_1, S_2, T)$ by applying a *sigmoid* non-linearity. We build two completely separate models, one for *pneu* and one for inf, and optimize the neural network weights using maximum-likelihood estimation with a binary cross-entropy loss. As a vector representation for the text, we use *BioLORD* [17], which returns a 768-dimensional embedding of the text description. The tabular features are turned into a one-hot encoding, with 11 dimensions in total. Note that each symptom is encoded into a three-dimensional vector, to be able to model the case where the symptom is unobserved, next to its two possible classes (*yes*/*no*). We also experimented with including pairwise, three-way and four-way interactions of background and symptom representations at the input, which adds another 180 dimensions. See Appendix D.1 for the final hyperparameter configuration.

Inference The model is trained to maximize the likelihood $\mathcal{P}(D_i \mid B, S_0, S_1, S_2, T)$, so we can directly obtain this probability as an output to the model when we input a test sample. To get a prediction for $\mathcal{P}(D_i \mid B, S_0, S_1, S_2)$, we replace the text at the input by an empty string (simply “”) and use its *BioLORD* embedding. Note that the model is equipped to deal with this, since these empty texts occur in 1/3 of the training data as well. Finally, to get a prediction for $\mathcal{P}(D_i \mid B, T)$, we set all symptoms to unobserved and use their corresponding one-hot encoding at the input of the model, instead of the original encoding.

C.3 Bayesian Network with Text Generator (`BN-gen-text`)

The joint probability distribution modeled by the Bayesian network with text generator is given in Eq. 6. We parameterize each conditional distribution as a Bernoulli distribution with one trainable parameter per conditional parent configuration, except for the text T, which fits a Gaussian distribution to the text embeddings as explained in Sect. 4.1. We then learn all trainable parameters using maximum likelihood estimation where the likelihood $\mathcal{P}(B, D_0, D_1, S_0, S_1, S_2)$ (Eq. 6 without factor $\mathcal{P}(T \mid D_0, D_1, S_0, S_1, S_2)$) is maximized based on the training data. This essentially comes down to filling in the CPTs like in a normal Bayesian network.

$$\begin{aligned}\mathcal{P}(B, D_0, D_1, S_0, S_1, S_2, T) = &\mathcal{P}(B)\mathcal{P}(D_0 \mid B)\mathcal{P}(D_1 \mid B)\mathcal{P}(S_0 \mid D_0)\\ &\mathcal{P}(S_1 \mid D_0, D_1)\mathcal{P}(S_2 \mid D_1)\mathcal{P}(T \mid D_0, D_1, S_0, S_1, S_2) \quad (6)\end{aligned}$$

Note that the generative model can easily deal with missing data: if the symptoms are unobserved, only the parameters for $\mathcal{P}(B)$, $\mathcal{P}(D_0 \mid B)$ and $\mathcal{P}(D_1 \mid B)$ are updated. Similarly, samples where the text is missing still contribute to the learned CPTs, while $\mathcal{P}(T \mid D_0, D_1, S_0, S_1, S_2)$ is fitted separately to the observed text embeddings only.

Equations 7, 8 and 9 show how we calculate the posterior likelihood for the diagnoses through Bayesian inference, for different sets of evidence. Note that we write sums for clarity, but strictly speaking marginalization over T is done by integration over the normally distributed text embedding variable.

$$\begin{aligned}\mathcal{P}(D_i \mid B, S_0, S_1, S_2) &= \frac{\sum_{D_{1-i},T} \mathcal{P}(B, D_0, D_1, S_0, S_1, S_2, T)}{\sum_{D_0,D_1,T} \mathcal{P}(B, D_0, D_1, S_0, S_1, S_2, T)}\\ &= \frac{\begin{array}{r}\sum_{D_{1-i}} \mathcal{P}(B)\mathcal{P}(D_0 \mid B)\mathcal{P}(D_1 \mid B)\mathcal{P}(S_0 \mid D_0)\mathcal{P}(S_1 \mid D_0, D_1)\\ \mathcal{P}(S_2 \mid D_1) \sum_T \mathcal{P}(T \mid D_0, D_1, S_0, S_1, S_2)\end{array}}{\begin{array}{r}\sum_{D_0,D_1} \mathcal{P}(B)\mathcal{P}(D_0 \mid B)\mathcal{P}(D_1 \mid B)\mathcal{P}(S_0 \mid D_0)\mathcal{P}(S_1 \mid D_0, D_1)\\ \mathcal{P}(S_2 \mid D_1) \sum_T \mathcal{P}(T \mid D_0, D_1, S_0, S_1, S_2)\end{array}}\\ &= \frac{\sum_{D_{1-i}} \mathcal{P}(B)\mathcal{P}(D_0 \mid B)\mathcal{P}(D_1 \mid B)\mathcal{P}(S_0 \mid D_0)\mathcal{P}(S_1 \mid D_0, D_1)\mathcal{P}(S_2 \mid D_1)}{\sum_{D_0,D_1} \mathcal{P}(B)\mathcal{P}(D_0 \mid B)\mathcal{P}(D_1 \mid B)\mathcal{P}(S_0 \mid D_0)\mathcal{P}(S_1 \mid D_0, D_1)\mathcal{P}(S_2 \mid D_1)} \quad (7)\end{aligned}$$

$$\begin{aligned}\mathcal{P}(D_i \mid B, S_0, S_1, S_2, T) &= \frac{\sum_{D_{1-i}} \mathcal{P}(B, D_0, D_1, S_0, S_1, S_2, T)}{\sum_{D_0,D_1} \mathcal{P}(B, D_0, D_1, S_0, S_1, S_2, T)}\\ &= \frac{\begin{array}{r}\sum_{D_{1-i}} \mathcal{P}(B)\mathcal{P}(D_0 \mid B)\mathcal{P}(D_1 \mid B)\mathcal{P}(S_0 \mid D_0)\mathcal{P}(S_1 \mid D_0, D_1)\\ \mathcal{P}(S_2 \mid D_1)\mathcal{P}(T \mid D_0, D_1, S_0, S_1, S_2)\end{array}}{\begin{array}{r}\sum_{D_0,D_1} \mathcal{P}(B)\mathcal{P}(D_0 \mid B)\mathcal{P}(D_1 \mid B)\mathcal{P}(S_0 \mid D_0)\mathcal{P}(S_1 \mid D_0, D_1)\\ \mathcal{P}(S_2 \mid D_1)\mathcal{P}(T \mid D_0, D_1, S_0, S_1, S_2)\end{array}} \quad (8)\end{aligned}$$

$$\begin{aligned}\mathcal{P}(D_i \mid B, T) &= \frac{\sum_{D_{1-i},S_0,S_1,S_2} \mathcal{P}(B, D_0, D_1, S_0, S_1, S_2, T)}{\sum_{D_0,D_1,S_0,S_1,S_2} \mathcal{P}(B, D_0, D_1, S_0, S_1, S_2, T)}\\ &= \frac{\begin{array}{r}\sum_{D_{1-i},S_0,S_1,S_2} \mathcal{P}(B)\mathcal{P}(D_0 \mid B)\mathcal{P}(D_1 \mid B)\mathcal{P}(S_0 \mid D_0)\mathcal{P}(S_1 \mid D_0, D_1)\\ \mathcal{P}(S_2 \mid D_1)\mathcal{P}(T \mid D_0, D_1, S_0, S_1, S_2)\end{array}}{\begin{array}{r}\sum_{D_0,D_1,S_0,S_1,S_2} \mathcal{P}(B)\mathcal{P}(D_0 \mid B)\mathcal{P}(D_1 \mid B)\mathcal{P}(S_0 \mid D_0)\mathcal{P}(S_1 \mid D_0, D_1)\\ \mathcal{P}(S_2 \mid D_1)\mathcal{P}(T \mid D_0, D_1, S_0, S_1, S_2)\end{array}} \quad (9)\end{aligned}$$

C.4 Bayesian Network with Text Discriminator (`BN-discr-text`)

The joint probability distribution modeled by the Bayesian network with text discriminator is given by Eq. 10. Like before, $\mathcal{P}(B)$ is parameterized as a Bernoulli distribution with one trainable parameter. All other conditional probability distributions are parameterized by one neural text discriminator per conditional parent configuration, resulting in 2 classifiers per conditional distribution, except for $\mathcal{P}(S_1 \mid D_0, D_1, T)$, which has 4 (due to *cough* having both *pneu* and *inf* as a parent). Each text classifier is modeled as a discriminative feed-forward neural network that takes a vector representation of the text as an input (once again, we use 768-dimensional *BioLORD* embeddings), transforming it into a one-dimensional representation which is then turned into a prediction for $\mathcal{P}(X \mid \mathcal{Y} = y, T)$ by applying a *sigmoid* non-linearity. Here, X represents the child variable (either D_0, D_1, S_0, S_1 or S_2), while $\mathcal{Y}$ represents the parent variable (either B, D_0, D_1 or $\{D_0, D_1\}$) taking on the configuration y. The Bernoulli parameter and neural network weights are trained by jointly maximizing the likelihood in Eq. 10 over the training set.

$$\begin{aligned}\mathcal{P}(B, D_0, D_1, S_0, S_1, S_2 \mid T) = \mathcal{P}(B)\mathcal{P}(D_0 \mid B, T)\mathcal{P}(D_1 \mid B, T)\mathcal{P}(S_0 \mid D_0, T)\\ \mathcal{P}(S_1 \mid D_0, D_1, T)\mathcal{P}(S_2 \mid D_1, T)\end{aligned} \tag{10}$$

Note that the discriminative model can easily deal with missing data: if the symptoms are unobserved, only the parameters for $\mathcal{P}(B)$, $\mathcal{P}(D_0 \mid B, T)$ and $\mathcal{P}(D_1 \mid B, T)$ are updated. When the text is unobserved, we input the *BioLORD* embedding for an empty text into each classifier. For that particular input, the classifiers will simply learn the co-occurrence of child and parent values in the 1/3 of the training data where text is empty, their outputs essentially mimicking the CPTs in a normal Bayesian network (like in the **`BN`** baseline model).

Equations 11 and 12 show how we calculate the posterior likelihood for the diagnoses through Bayesian inference, taking the symptoms and text (empty or not) as evidence. $\mathcal{P}(D_i \mid B, T)$ can simply be taken directly as the output of the relevant diagnosis classifier.

$$\begin{aligned}\mathcal{P}(D_i \mid B, S_0, S_1, S_2, T) &= \frac{\sum_{D_{1-i}} \mathcal{P}(B, D_0, D_1, S_0, S_1, S_2 \mid T)}{\sum_{D_0, D_1} \mathcal{P}(B, D_0, D_1, S_0, S_1, S_2 \mid T)}\\ &= \frac{\begin{array}{r}\sum_{D_{1-i}} \mathcal{P}(B)\mathcal{P}(D_0 \mid B, T)\mathcal{P}(D_1 \mid B, T)\mathcal{P}(S_0 \mid D_0, T)\\ \mathcal{P}(S_1 \mid D_0, D_1, T)\mathcal{P}(S_2 \mid D_1, T)\end{array}}{\begin{array}{r}\sum_{D_0, D_1} \mathcal{P}(B)\mathcal{P}(D_0 \mid B, T)\mathcal{P}(D_1 \mid B, T)\mathcal{P}(S_0 \mid D_0, T)\\ \mathcal{P}(S_1 \mid D_0, D_1, T)\mathcal{P}(S_2 \mid D_1, T)\end{array}}\end{aligned} \tag{11}$$

$$\mathcal{P}(D_i \mid B, S_0, S_1, S_2) = \mathcal{P}(D_i \mid B, S_0, S_1, S_2, T = \text{“”}) \tag{12}$$

D Empirical Results and Analysis

D.1 Training, Hyperparameter Tuning and Evaluation

All models, except the baselines **BN** and **BN++**, have multiple hyperparameters to tune. We optimized these separately for each model using a train-validation split on the train set (3200/800 sample split out of 4000 samples in total), choosing the hyperparameters that maximize the average precision of $\mathcal{P}(D_i|B, S_0, S_1, S_2, T)$ on the validation set. The full implementation can be found in our Github repository: https://github.com/prabaey/bn-text.

- **FF-discr-text** We optimized the number of epochs, the number of layers (including their width), the batch size, learning rate and weight decay of the Adam optimizer, dropout and whether to include interaction features at the input or not. We optimized these hyperparameters separately for the *pneu* and *inf* classifier. For the *pneu* classifier, the final configuration we landed on was the following: 200 epochs, 2 layers (dimensions 768 → 256 → 1, with a ReLU activation in the middle), batch size 256, learning rate 1e−2, weight decay 1e−3, dropout of 70% in every layer and no interaction features. For the *inf* classifier, the optimal settings were the same, except that it had 1 layer (dimensions 768 → 1). To make up for the lower complexity of the model (and limited ability to mix features in a single layer), it proved optimal to include the interaction features at the input of this classifier.
- **BN-discr-text** To make for a fair comparison, we used the same layer and dropout configurations that were chosen after tuning the **FF-discr-text** model for the $\mathcal{P}(D_0|B, T)$ classifier (*pneu*) and $\mathcal{P}(D_1|B, T)$ classifier (*inf*). The symptom classifiers already achieved perfect performance with only 1 layer (dimensions 768 → 1) and without dropout, so we kept these settings. We again used the Adam optimizer to learn the neural weights for all classifiers, with learning rate 1e−2 and weight decay 1e−3. We used a separate learning rate of 0.05 (without weight decay) for learning the $\mathcal{P}(B)$ distribution, which is modeled with a single Bernoulli parameter. Other hyperparameters were also chosen in accordance with the **FF-discr-text** model: 200 epochs and batch size 256.
- **BN-gen-text** For learning the conditional probability table parameters in the Bayesian network, we used an Adam optimizer with a learning rate of 0.05 and no weight decay. We trained for 15 epochs with a batch size of 256. Hyperparameter α, which regularizes the covariance matrix in Eq. 1, was found to be optimal at 0.85. We use the same α for all 32 Gaussians.

We trained all models with their optimal hyperparameter configurations over the train set of 4000 samples. We repeated this process 5 times, each time with a different initialization seed (except for the **BN** baseline, which is deterministic). For each trained model, we calculated the three posterior diagnosis probabilities for all 1000 samples in the test set. We then obtained the average precision (area under the precision-recall curve) by comparing each prediction to the known label for the diagnosis. We report average precision rather than area under the ROC curve (another metric often used to assess classification performance), since the former is better suited to evaluate predictive performance in extremely imbalanced datasets [3], which is the case for pneumonia.[1] Furthermore, balancing precision and recall (catching as many cases of pneumonia as possible without including too many false positives) describes the diagnostic task of the GP in the practical use case well. Table 1 in the main text shows the results for the prediction of *pneu*, while Table 2 shows these results for the prediction of *inf*.

Table 2. Average precision over test set of three posterior probabilities for the diagnosis *inf*, each taking a different set of evidence into account (various combinations of background, symptoms and text). We show mean (± std) over 5 initialization seeds.

Model	Average precision for *inf*		
	$P(D_1 \mid B, S_0, S_1, S_2, T)$	$P(D_1 \mid B, S_0, S_1, S_2)$	$P(D_1 \mid B, T)$
BN	-	0.8884 (± 0.0000)	-
BN^{++}	-	0.9009 (± 0.0000)	-
FF-discr-text	0.9042 (± 0.0018)	0.8813 (± 0.0003)	0.8821 (± 0.0014)
BN-gen-text	0.7968 (± 0.0007)	0.8889 (± 0.0000)	0.7624 (± 0.0011)
BN-discr-text	0.9016 (± 0.0007)	0.8889 (± 0.0000)	0.8738 (± 0.0018)

D.2 Ablation Study

In designing the DAG for models **BN-gen-text** and **BN-discr-text**, we explicitly included an arc between each diagnosis and text. This modeling decision makes sense when one assumes the presence of some unknown and unobserved symptoms in the text. In this section, we investigate how the models would perform if these relations were left out. We first introduce our generative and discriminative ablated models, and then discuss the empirical results.

Ablated BN with Text Generator (BN-gen-text$^-$). We remove the arcs $D_0 \rightarrow T$ and $D_1 \rightarrow T$ from the **BN-gen-text** model shown in Fig. 2, forming the **BN-gen-text**$^-$ model. The text node now has only three parents (symptoms S_0, S_1 and S_2), meaning only 8 conditional Gaussians have to be fitted. Note that this means there are more text embeddings available to fit each Gaussian than there were for the **BN-gen-text** model. The new joint distribution modeled by this Bayesian network is shown in Eq. 13. Note that it differs from Eq. 6 only in its definition of the conditional text distribution. We train this model in the same way as before, with the hyperparameters described in Sect. D.1.

[1] We have a positive pneumonia label for only 34 out of 4000 samples in the training set and 14 out of 1000 samples in the test set.

$$\mathcal{P}(B, D_0, D_1, S_0, S_1, S_2, T) = \mathcal{P}(B)\mathcal{P}(D_0 \mid B)\mathcal{P}(D_1 \mid B)\mathcal{P}(S_0 \mid D_0) \\ \mathcal{P}(S_1 \mid D_0, D_1)\mathcal{P}(S_2 \mid D_1)\mathcal{P}(T \mid S_0, S_1, S_2) \quad (13)$$

Bayesian inference over the ablated DAG partially differs from inference over the original DAG. The calculation of both $\mathcal{P}(D_i \mid B, S_0, S_1, S_2)$ and $\mathcal{P}(D_i \mid B, T)$ incurs only minimal changes (just swap out $\mathcal{P}(T \mid D_0, D_1, S_0, S_1, S_2)$ for $\mathcal{P}(T \mid S_0, S_1, S_2)$ in Eqs. 7 and 9). However, the DAG shows that D_i is independent of T when all symptoms are known (no unblocked paths), meaning that $\mathcal{P}(D_i \mid B, S_0, S_1, S_2, T) = \mathcal{P}(D_i \mid B, S_0, S_1, S_2)$.

Ablated BN with Text Discriminator (`BN-discr-text`⁻). We remove the arcs $T \rightarrow D_0$ and $T \rightarrow D_1$ from the **`BN-discr-text`** model shown in Fig. 2, forming the **`BN-discr-text`**$^-$ model. This means that there are only 8 classifiers to be learned, as $\mathcal{P}(D_0 \mid B)$ and $\mathcal{P}(D_1 \mid B)$ can now be modeled as simple CPTs, just like $\mathcal{P}(B)$. The joint distribution for the ablated model is shown in Eq. 14. We train this model with the hyperparameters described in Sect. D.1, using a learning rate of 0.05 to learn the parameters of the CPTs for D_0, D_1 and B.

$$\mathcal{P}(B, D_0, D_1, S_0, S_1, S_2 \mid T) = \mathcal{P}(B)\mathcal{P}(D_0 \mid B)\mathcal{P}(D_1 \mid B)\mathcal{P}(S_0 \mid D_0, T) \\ \mathcal{P}(S_1 \mid D_0, D_1, T)\mathcal{P}(S_2 \mid D_1, T) \quad (14)$$

Again, Bayesian inference over the ablated DAG partially differs from inference over the original DAG. $\mathcal{P}(D_i \mid B, S_0, S_1, S_2, T)$ is calculated analogously to Eq. 11, but with $\mathcal{P}(D_i \mid B)$ instead of $\mathcal{P}(D_i \mid B, T)$. Finally, it is clear from the ablated DAG that the diagnoses are independent of the text if no symptoms are observed (all paths between D_i and T are blocked by unobserved colliders). Therefore $\mathcal{P}(D_i \mid B, T)$ equals $\mathcal{P}(D_i \mid B)$, meaning the **`BN-discr-text`**$^-$ model cannot extract any information from the text without any observed symptoms.

Table 3. Average precision over test set for the ablated text models, which do not explicitly include the relation between diagnoses and text.

Model	Average precision for *pneu*		
	$P(D_0 \mid B, S_0, S_1, S_2, T)$	$P(D_0 \mid B, S_0, S_1, S_2)$	$P(D_0 \mid B, T)$
`BN-gen-text`$^-$	0.0892 (± 0.0007)	0.0892 (± 0.0007)	0.0933 (± 0.0009)
`BN-discr-text`$^-$	0.1017 (± 0.0008)	0.1041 (± 0.0072)	0.0302 (± 0.0000)
Model	Average precision for *inf*		
	$P(D_1 \mid B, S_0, S_1, S_2, T)$	$P(D_1 \mid B, S_0, S_1, S_2)$	$P(D_1 \mid B, T)$
`BN-gen-text`$^-$	0.8889 (± 0.0000)	0.8889 (± 0.0000)	0.8914 (± 0.0002)
`BN-discr-text`$^-$	0.8065 (± 0.0004)	0.8889 (± 0.0000)	0.4441 (± 0.0000)

Analysis. Comparing the *pneu* portion of Table 3 with Table 1, we immediately note that performance drops dramatically in the ablated versions of both the generative and discriminative model. While $\mathcal{P}(D_0 \mid B, S_0, S_1, S_2)$ is still very similar to the **BN** baseline, including text in the prediction $\mathcal{P}(D_0 \mid B, S_0, S_1, S_2, T)$ now does not improve performance. Since we do not model the relation between diagnoses and text, the model can only extract information from the text *through* the three symptoms we explicitly include in the DAG: *dysp*, *cough* and *nasal*. Information regarding other useful symptoms, *pain* and *fever*, cannot be extracted.

While the **BN-gen-text**$^-$ model is able to extract the necessary information on the symptoms S_0, S_1 and S_2 from the text alone ($\mathcal{P}(D_0 \mid B, T) \sim \mathcal{P}(D_0 \mid B, S_0, S_1, S_2)$), **BN-discr-text**$^-$ performs abysmally when only text is included in the evidence. This comes as no surprise when we actually study the DAG: the diagnoses are independent of the text if no symptoms are observed. These independence assumptions do not match the reality we are trying to capture.

Comparing the *inf* portion of Table 3 with Table 2 shows lower performance of **BN-discr-text**$^-$ compared to **BN-discr-text** when only taking text into account ($\mathcal{P}(D_1 \mid B, T)$). Conversely, **BN-gen-text**$^-$ actually improves over **BN-gen-text** on both $\mathcal{P}(D_1 \mid B, S_0, S_1, S_2, T)$ and $\mathcal{P}(D_1 \mid B, T)$. Since the text node T now only has three parents instead of five, there's more text embeddings available to fit each conditional Gaussian. Combined with the fact that there is no additional information in the text that can help to predict *inf* anyway, modeling the direct relation between diagnosis and text will only result in a less reliable fit of the text distribution by the **BN-gen-text** model.

References

1. Ankan, A., Panda, A.: pgmpy: Probabilistic graphical models using Python. In: Proceedings of the 14th Python in Science Conference, pp. 6–11 (2015)
2. Chin-Yee, B., Upshur, R.: Clinical judgement in the era of big data and predictive analytics. J. Eval. Clin. Pract. **24**(3), 638–645 (2018)
3. Davis, J., Goadrich, M.: The relationship between precision-recall and roc curves. In: Proceedings of the 23rd International Conference on Machine Learning, p. 233–240 (2006)
4. Edye, E.O., et al.: Applying Bayesian networks to help physicians diagnose respiratory diseases in the context of COVID-19 pandemic. In: 2021 IEEE URUCON, pp. 368–371 (2021)
5. Ford, E., Carroll, J.A., Smith, H.E., Scott, D., Cassell, J.A.: Extracting information from the text of electronic medical records to improve case detection: a systematic review. J. Am. Med. Inform. Assoc. **23**(5), 1007–1015 (2016)
6. Gruppen, L.D.: Clinical reasoning: defining it, teaching it, assessing it, studying it. West J. Emerg. Med. **18**(1), 4–7 (2017)
7. Kyrimi, E., et al.: Bayesian networks in healthcare: what is preventing their adoption? Artif. Intell. Med. **116**, 102079 (2021)
8. Kyrimi, E., McLachlan, S., Dube, K., Neves, M.R., Fahmi, A., Fenton, N.: A comprehensive scoping review of Bayesian networks in healthcare: past, present and future. Artif. Intell. Med. **117**, 102108 (2021)

9. Manhaeve, R., Dumancic, S., Kimmig, A., Demeester, T., De Raedt, L.: DeepProbLog: neural probabilistic logic programming. In: Advances in Neural Information Processing Systems (NeurIPS), vol. 31 (2018)
10. Marra, G., Dumančić, S., Manhaeve, R., De Raedt, L.: From statistical relational to neurosymbolic artificial intelligence: a survey. Artif. Intell. **328**, 104062 (2024)
11. McLachlan, S., Dube, K., Hitman, G.A., Fenton, N.E., Kyrimi, E.: Bayesian networks in healthcare: distribution by medical condition. Artif. Intell. Med. **107**, 101912 (2020)
12. Mujtaba, G., et al.: Clinical text classification research trends: systematic literature review and open issues. Expert Syst. Appl. **116**, 494–520 (2019)
13. Norman, G.R., Monteiro, S.D., Sherbino, J., Ilgen, J.S., Schmidt, H.G., Mamede, S.: The causes of errors in clinical reasoning: cognitive biases, knowledge deficits, and dual process thinking. Acad. Med. **92**(1), 23–30 (2017)
14. Ouyang, L., Wu, J., Jiang, X., Almeida, D., et al.: Training language models to follow instructions with human feedback. In: Advances in Neural Information Processing Systems, vol. 35, pp. 27730–27744 (2022)
15. Peiffer-Smadja, N., Rawson, T., Ahmad, R., Buchard, A., et al.: Machine learning for clinical decision support in infectious diseases: a narrative review of current applications. Clin. Microbiol. Infect. **26**(5), 584–595 (2020)
16. Price, S.J., Stapley, S.A., Shephard, E., Barraclough, K., Hamilton, W.T.: Is omission of free text records a possible source of data loss and bias in clinical practice research datalink studies? A case–control study. BMJ Open **6**(5), e011664 (2016)
17. Remy, F., Demuynck, K., Demeester, T.: BioLORD: semantic textual representations fusing LLM and clinical knowledge graph insights. arXiv preprint (2023)
18. Rotmensch, M., Halpern, Y., Tlimat, A., Horng, S., Sontag, D.: Learning a health knowledge graph from electronic medical records. Sci. Rep. **7**(1), 5994 (2017)
19. Sterckx, L., Vandewiele, G., Dehaene, I., Janssens, O., et al.: Clinical information extraction for preterm birth risk prediction. J. Biomed. Inform. **110**, 103544 (2020)
20. Strauss, S.E., Glasziou, P., Richardson, W.S., Haynes, R.B.: Evidence-based medicine. In: How to Practice and Teach EBM, vol. 5. Elsevier, Amsterdam (2018)
21. Yanase, J., Triantaphyllou, E.: A systematic survey of computer-aided diagnosis in medicine: past and present developments. Expert Syst. Appl. **138**, 112821 (2019)
22. Yazdani, S., Hoseini Abardeh, M.: Five decades of research and theorization on clinical reasoning: a critical review. Adv. Med. Educ. Pract. **10**, 703–716 (2019)
23. Yazdani, S., Hosseinzadeh, M., Hosseini, F.: Models of clinical reasoning with a focus on general practice: a critical review. J. Adv. Med. Educ. Prof. **5**(4), 177–184 (2017)
24. Ye, Y., Tsui, F., Wagner, M., Espino, J., Li, Q.: Influenza detection from emergency department reports using natural language processing and Bayesian network classifiers. J. Am. Med. Inform. Assoc. **21**(5), 815–823 (2014)
25. Zhang, D., Yin, C., Zeng, J., Yuan, X., Zhang, P.: Combining structured and unstructured data for predictive models: a deep learning approach. BMC Med. Inform. Decis. Mak. **20**(1), 280 (2020)

Empowering Language Model with Guided Knowledge Fusion for Biomedical Document Re-ranking

Deepak Gupta(✉) and Dina Demner-Fushman

LHNCBC, National Library of Medicine, National Institutes of Health,
Bethesda, MD, USA
{deepak.gupta,dina.demner-fushman}@nih.gov

Abstract. Pre-trained language models (PLMs) have proven to be effective for document re-ranking task. However, they lack the ability to fully interpret the semantics of biomedical and healthcare queries and often rely on simple patterns for retrieving documents. To address this challenge, we propose an approach that integrates medical knowledge into PLMs to guide the model toward effectively capturing information from external sources and retrieving the correct documents. We performed comprehensive experiments on two biomedical datasets and an open-domain dataset. We demonstrate the capability of the proposed mutual information-based feature fusion technique by comparing it with the existing feature fusion techniques. Our extensive experiments on multiple datasets show that our proposed approach significantly improves vanilla PLMs and other existing approaches for document re-ranking task in the biomedical/clinical domain.

Keywords: Information Retrieval · Knowledge Graph

1 Introduction

Retrieving relevant information in response to a query involves considering both the explicit constraints indicated in the textual contents of the query and the implicit knowledge about the domain of interest. Large pre-trained language models [3,15] became a foundation for most modern information retrieval (IR) systems. While these models have acquired the ability to implicitly encode broad world knowledge and have achieved significant performance on a variety of benchmark tasks, they fall short when provided with examples that are distributionally distinct from those they were fine-tuned on. This limitation of PLMs is further amplified in the biomedical/clinical setting, where **(i)** there is a high degree of variability in the form of synonyms and abbreviated words and **(ii)** retrieval of relevant information is dependent on understanding the focus/intent of the query. In the example shown in Table 1, the query context is neither explicitly stated in the gold-standard document, nor does it contain one salient term ('*chromosome 13*'). It requires domain knowledge to infer that a type of *omodysplasia*

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 251–260, 2024.
https://doi.org/10.1007/978-3-031-66538-7_25

is an *"autosomal recessive disorder"* caused by mutation in a gene on one of the first 22 non-sex chromosomes. In such cases, which require domain knowledge to correctly retrieve relevant documents, both BM25 and MonoT5 fail. These findings highlight that PLMs lack semantic interpretation of queries and oftentimes depend on naïve patterns to retrieve information rather than using more structured reasoning that effectively amalgamates information provided in the context with external knowledge. In the past, there have been research efforts [5,23] to fuse domain knowledge in LMs, yet so far, to the best of our knowledge, there has been no exploration towards integrating external knowledge in neural IR, both in open domain and much-needed biomedical/clinical domain.

Table 1. Top retrieved documents using BM25 and MonoT5 models for the query: "*What rare disease is associated with a mutation in the GPC6 gene on chromosome 13?*" along with the gold-standard document. Lexical and semantic matches considering context are shown in blue and pink, respectively. The highlighted texts in green indicate domain knowledge needed to retrieve the correct document. For a comprehensive view, access the online version where the tables are displayed in color.

Top Retrieved Document (BM25)	**Top Retrieved Document (MonoT5)**	**Gold Document**
..We report the construction of a high-resolution 4 Mb sequence-ready BAC/PAC contig of the GPC5/GPC6 gene cluster on chromosome region 13q32.	The human gamma-sarcoglycan gene was mapped to chromosome 13q12, and deletions that alter its reading frame were identified in three families and one of four sporadic cases of SCARMD.	.. The proband had normal molecular analysis of the glypican 6 gene (GPC6), which was recently reported as a candidate for autosomal recessive omodysplasia. Mild rhizomelic shortening of the lower extremities has not been previously reported...

To address the aforementioned issues, in this work, we propose GraphMonoT5, an effective approach that fuses the external knowledge into the pretrained language model for the document retrieval task. The proposed GraphMonoT5 is built upon the encoder-decoder T5 model, and the T5 encoder layer is complemented with the graph neural network (GNN). The former takes query and document as input and later is used to reason over the underlying knowledge graph (KG) with entities as nodes and relationships between them as edges. With the use of mutual information-based interaction representations, we develop a strategy to effectively fuse the language and graph representation and allow a two-way exchange of information between the text and graph modalities.

2 Methodology

Background. Our proposed re-ranking approach GraphMonoT5 is based on the MonoT5 model that utilizes the encoder-decoder based T5 [15] model to calculate a relevance score that provides a quantitative indication of the degree to which a candidate document d is pertinent to a query q. The input prompt to the MonoT5 model is:

$$\text{Query: } [q] \quad \text{Document: } [d] \text{ Relevant:} \tag{1}$$

The MonoT5 model is fine-tuned to generate the words "true" for relevant or "false" for the documents non-relevant to the query. During inference, the candidate documents are re-ranked based on the probability of the "true" token.

2.1 Proposed Model

Our proposed GraphMonoT5 model is the result of the augmentation of the PLM with the graph reasoning modules over KG for effectively re-ranking the candidate documents against the query. We describe the KG construction and KG-enriched ranking in the following subsections:

Knowledge Graph Construction. The knowledge graph is a multi-relational graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with entity nodes $\mathcal{V}$ and edges $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{R} \times \mathcal{V}$ that connect nodes in $\mathcal{V}$ with the set of relations $\mathcal{R}$. Given a query-document pair (q, d), we link the entities (*cf.* Appendix C.1 [1]) mentioned in the query and document to the KG $\mathcal{G}$. The nodes corresponding to query q and document d are denoted by $\mathcal{V}_q \subseteq \mathcal{V}$ and $\mathcal{V}_d \subseteq \mathcal{V}$, respectively. The total nodes of the query-document pair are denoted by $\mathcal{V}_{q,d} = \mathcal{V}_q \cup \mathcal{V}_d$. Since the KG $\mathcal{G}$ can include millions of nodes and edges, only a subgraph $\mathcal{G}_{q,d} = (\mathcal{V}_{q,d}, \mathcal{E}_{q,d})$ of the KG $\mathcal{G}$ which contains all the nodes on the 2-hop paths between nodes in $\mathcal{V}_{q,d}$ is considered for the query-document pair.

KG-Enriched Seq2Seq Ranking. Our KG-enriched seq2seq ranking approach consists of **(a)** R layers `T5-encoder` model to encode the language context, **(b)** graph neural network to model the subgraph of the query-document pair, **(c)** S layers language-graph interaction component to fuse the language and graph representations, and **(d)** T5-decoder model to predict the query-document relevance score. Following, [25], we use an interaction token t_{int} and interaction node n_{int} to pass the information across the language and graph modalities. In contrast to [25], we introduced a mutual information-based fusion technique that combines new information from the PLM and knowledge graph by eliminating redundancy. The interaction token t_{int} is prepended to the token sequence $\{t_1, t_2, \ldots, t_N\}$ of query-document pair (q, d) (*cf.* Eq. 1) and n_{int} is connected to all the nodes in $\mathcal{G}_{q,d}$, allowing the information flow amongst the nodes in $\mathcal{G}_{q,d}$ and being the representative node of $\mathcal{G}_{q,d}$.

Language Representation: Given the token sequence $\mathcal{T} = \{t_{int}, t_1, t_2, \ldots, t_N\}$, first we pass the sequence $\mathcal{T}$ into the first layer of the `T5-encoder` [15] to obtain the hidden state representations $H^1 = \{h^1_{int}, h^1_1, h^1_2, \ldots, h^1_N\} \in \mathcal{R}^{(N+1) \times d_l}$, where d_l is the dimension of the hidden state. Hidden state representation H^l at l^{th} layer is passed to the $(l+1)^{th}$ layer of `T5-encoder` to encode and obtain the representation H^{l+1}. Following this, we extract the representation from `T5-encoder` for $l = 1, 2, \ldots, R$:

$$h^{l+1}_{int}, h^{l+1}_1, \ldots, h^{l+1}_N = \texttt{T5-encoder}(h^l_{int}, h^l_1, \ldots, h^l_N) \tag{2}$$

To fuse the language and graph representations, we also extract the hidden state representation from an additional S layers of `T5-encoder`; however, at layer l the interaction token representation h^l_{int} is fused with the interaction node representation (to be discussed shortly) to amalgamate the knowledge feature with the language model feature.

Graph Representation: Given the query-document pair sub-graph $\mathcal{G}_{q,d} = (\mathcal{V}_{q,d}, \mathcal{E}_{q,d})$ with nodes $\{n_{int}, n_1, n_2, \ldots, n_M\}$, we first compute the node embeddings $U^1 = \{u^1_{int}, u^1_1, u^1_2, \ldots, u^1_M\} \in \mathcal{R}^{(M+1)\times d_g}$ using the pre-trained knowledge graph embeddings (Appendix C.2 [1]). We utilize the graph attention network [19] to compute the node representation by propagating the information across the nodes in the subgraph $\mathcal{G}_{q,d}$. The subgraph node representation U^l at l^{th} layer of GNN is passed to the $(l+1)^{th}$ layer of GNN to encode and obtain the representation U^{l+1}. Following this, we extract the representation from GNN for $l = 1, 2, \ldots, S$:

$$u^{l+1}_{int}, u^{l+1}_1, \ldots, u^{l+1}_M = \mathsf{GNN}(u^l_{int}, u^l_1, \ldots, u^l_M) \tag{3}$$

Language-Graph Interaction: On a given layer $l \in S$, we aim to effectively fuse the modalities by using the interaction token representation h^l_{int} and interaction node representation u^l_{int}. Towards this, first, we obtain the fused representation $x^l = f(h^l_{int} \oplus u^l_{int})$ with a two-layer feed-forward network f. The fused representation x^l may contain redundant information. To overcome this issue, we introduce mutual information (MI)-based feature fusion that aims to minimize the MI $\mathcal{I}(x^l; z^l)$ between the compressed encoded representation z^l and the concatenated representation x^l. Formally given two random variables x^l and z^l, their MI is defined as follows:

$$\begin{aligned}\mathcal{I}(x^l; z^l) &= D_{KL}(p(x^l, z^l)||p(x^l)p(z^l)) \\ &\leq \alpha \mathbb{E}_{z^l \sim p(z^l|x^l)}[D_{KL}(p(z^l|x^l)||q(z^l))] \\ &\leq \alpha M(x^l; z^l)\end{aligned} \tag{4}$$

where, α is a constant and D_{KL} denotes the KL divergence (proof in Appendix B.2 [1]). We model the $p(z^l|x^l)$ using a parameterized Gaussian distribution $\mathcal{N}(\mu^l_z, \Sigma^l_z)$ with mean μ^l_z and variance Σ^l_z. To compute the gradients through random variables, we follow the reparametrization trick [9] with standard normal distribution $\epsilon \sim \mathcal{N}(0, I)$ to calculate $z^l = \mu^l_z + \Sigma^l_z \epsilon$. Later, we split z^l into the $\tilde{h}^l_{int}$ and $\tilde{u}^l_{int}$ for further computation of the token and node, respectively. With the virtue of the Transformer network and GNN, the fused representation is mixed with the remaining tokens and nodes of the subgraph. The graph-augmented representations from the KG-enriched `T5-encoder` are passed to the `T5-decoder` to predict the query-document relevance score as discussed in Eq. 1.

Network Training: The network is trained by maximizing the log-likelihood of the document given the query and minimizing the mutual information on each layer of the KG-enriched `T5-encoder`. Formally the training objective J is

$$J = p(y|\mathcal{T}) - \frac{\alpha}{S}\sum_{l=1}^{S} M(x^l; z^l) \tag{5}$$

where $y \in \{\text{'true'}, \text{'false'}\}$ is the predicted token from T5 model given the input token sequence $\mathcal{T}$.

3 Experimental Setups

Datasets and Knowledge Sources. We evaluated our proposed GraphMonoT5 model on two existing biomedical datasets: BioASQ8B [12] and TREC-COVID [20]. Additionally, we evaluated our approach on an open-domain HotPotQA [22] that includes PubMed and Wikipedia articles, which also contain health-related information. For biomedical domains, we train the model on the training collection of the BioASQ8B dataset, the network hyper-parameters are tuned on batch four of BioASQ7B test collection. Performance is reported on the five different test collections (B1, B2, B3, B4, and B5) each of 100 queries of BioASQ8B and TREC-COVID datasets. We utilized *ConceptNet*[1], an open-domain knowledge graph, to extract knowledge for HotPotQA dataset and biomedical knowledge graph from [25] that was developed by integrating the Unified Medical Language System (UMLS)[2] and DrugBank knowledge sources to extract knowledge from BioASQ datasets. The detailed statistics of the datasets and knowledge graph are shown in Table 2.

Table 2. Statistics of the datasets used in the experiments. For TREC-COVID, the performance is evaluated in zero-shot settings on the model built upon the BioASQ8B training dataset.

Datasets	# Query-docs	#Dev	#Test	Corpus	Nodes	Edges
BioASQ8B	32,916	100	500	14,914,602	9,958	44,561
TREC-COVID	–	–	50	171,332	–	–
HotPotQA	170,000	5,447	7,405	5,233,329	799,273	2,487,810

Evaluation. Following the existing works on BioASQ8B, we evaluated the performance of the models using Mean Average Precision (MAP), Recall@100 (R@100), and normalized cumulative discount gain (nDCG@10). We used the official BioASQ script[3] to compute MAP values, and Pytrec_eval to report the nDCG@10 and Recall@100 score. Following [18], we report the Capped Recall@100 score for the TREC-COVID dataset.

4 Results and Analysis

Results. Table 3 demonstrates that the GraphMonoT5 model equipped with knowledge-graph outperforms the existing approaches on BioASQ8B, TREC-COVID, and HotPotQA test datasets. Since the TREC-COVID dataset does not contain the training set, we evaluated the model trained on the BioASQ8B dataset on the test set of TREC-COVID in a zero-shot setting. We have also

[1] https://conceptnet.io/.

[2] https://www.nlm.nih.gov/research/umls/index.html.

[3] https://github.com/BioASQ/Evaluation-Measures.

provided a performance comparison of our proposed approach with the best systems of the BioASQ8 challenge, and recent work of [11] in Table 4. In the BioASQ8B, there are five different test sets (B1, B2, B3, B4, and B5) periodically released. We computed the performance on each of these test sets and reported the results (*cf.* Table 4) in terms of MAP.

Table 3. Performance comparison of our proposed method with the existing approaches on respective datasets. R@100 refers to the Recall@100. The first block of the results is taken from [18]

Models	BioASQ8B		TREC-COVID		HotPotQA	
	R@100	nDCG@10	R@100	nDCG@10	R@100	nDCG@10
DeepCT [2]	0.699	0.407	0.347	0.406	0.731	0.503
SPARTA [26]	0.351	0.351	0.409	0.538	0.651	0.492
DPR [6]	0.256	0.127	0.212	0.332	0.591	0.391
ANCE [21]	0.463	0.306	0.457	0.654	0.578	0.456
TAS-B [4]	0.579	0.383	0.387	0.481	0.728	0.584
GenQ [18]	0.627	0.398	0.456	0.619	0.673	0.534
ColBERT [8]	0.645	0.474	0.464	0.677	0.748	0.593
BM25 [17]	0.745	0.488	0.508	0.688	0.763	0.602
MonoT5 [13]	0.745	0.489	0.508	0.685	0.763	0.648
Proposed (GraphMonoT5)	0.745	**0.520**	0.508	**0.701**	0.763	**0.667**
w/o **MI Fusion**	0.745	0.499	0.508	0.683	0.763	0.637

Quantitative Analysis: To analyze the role of mutual information based objective function, we trained the model with only cross-entropy loss and observed the decrements of 2.1, 1.8, and 3.0 nDCG@10 points over the cross-entropy with MI objective on BioASQ8B, TREC-COVID, and HotPotQA dataset respectively. We have also provided (*cf.* Fig. 1) the comparison of the approaches in terms of MAP, which shows that the GraphMonoT5 method with mutual information fusion outperforms the MonoT5 and concatenation-based fusion on BioASQ8B and HotPotQA datasets. Compared to one of the best-performing systems [7] of BioASQ8B, our proposed approach shows an absolute improvement of 4.23 points in MAP score on BioASQ8B test sets. We also see an absolute improvement of 3.24 points in MAP score on BioASQ8B test sets compared to the [14] work. Our approach outperformed the recent work of [11] by 6.82 points MAP score on the BioASQ8B test sets.

Influence of the KG: We performed an ablation study to assess the role of KG in our proposed approach. With GraphMonoT5, we observed a significant ($p < 0.05$, using bootstrap test) improvement of 3.1, 1.6, and 1.9 nDCG@10 points over the vanilla MonoT5 model on BioASQ8B, TREC-COVID, and HotPotQA datasets, respectively. Furthermore, compared to BM25, we observed an improvement of 3.2, 1.3, and 6.5 nDCG@10 points on respective datasets. The results allow for two important claims **(1)** knowledge-enriched PLMs help to

Table 4. Comparison of the proposed method with the state-of-the-art approaches on BioASQ8B test batches in terms of MAP score.

Methods	B1	B2	B3	B4	B5	Mean
PA [7]	0.3346	0.3304	0.4351	0.3600	**0.4825**	0.3885
AUEB [14]	0.3359	0.3181	0.4510	0.4163	0.4657	0.3974
P-DPR [11]	0.3002	0.3131	0.3979	0.4218	0.3799	0.3626
Proposed (GraphMonoT5)	**0.3906**	**0.3943**	**0.4697**	**0.5190**	0.4168	**0.4308**

Table 5. Performance Comparison (in terms of nDCG@10) of our proposed fusion strategy with the existing fusion approaches on benchmark datasets.

Approach	BioASQ8B	TREC-COVID	HotPotQA
GraphMonoT5 w/ MAG [16]	0.454	0.599	0.521
GraphMonoT5 w/ LMF [10]	0.479	0.656	0.509
GraphMonoT5 w/ TFN [24]	0.492	0.642	0.484
GraphMonoT5 w/ MI Fusion (Ours)	0.52	0.701	0.667

re-rank the documents more accurately compared to the vanilla PLMs and **(2)** mutual information-based knowledge-fusion is an appropriate strategy to fuse the language and graph information.

Comparisons with Existing Fusion Techniques: We performed extensive experiments to compare the performance of our proposed MI-based knowledge fusion strategy with the existing works on multimodal fusion. Towards this, we replaced the proposed MI-based knowledge fusion with the MAG fusion introduced in [16]. Similarly, we also performed the experiments with LMF [10] and TFN [24] fusion techniques. Table 5 reports the results on BioASQ8B, TREC-COVID, and HotPotQA datasets, comparing our proposed fusion technique with MAG, LMF, and TFN fusion techniques in terms of nDCG@10. Compared to MAG, our proposed fusion technique shows an improvement of 6.6, 10.2, and 14.6 points nDCG@10 on BioASQ8B, TREC-COVID, and HotPotQA datasets respectively. For the BioASQ8B, we obtained the best results of 0.492 nDCG@10, which is 2.8 points above the best pre-existing fusion technique TFN. Similarly, for the HotPOTQA dataset, our nDCG@10 score outperforms the second-best MAG fusion technique by 0.521. These comparisons confirm that the proposed knowledge fusion technique outperforms the existing fusion techniques on open-domain and biomedical-domain datasets.

Open vs. Biomedical Domain Performance: We also present a comparative analysis of the open and biomedical domain performance. Towards this, we evaluated the performance of the MonoT5, proposed GraphMonoT5, and GraphMonoT5 without MI-based technique (language-knowledge concatenation-based fusion) on BioASQ8B (biomedical) and HotPotQA (open) datasets in terms of MAP@k (k = 5, 10 and 20). It is observed from Fig. 1 that performance of the system

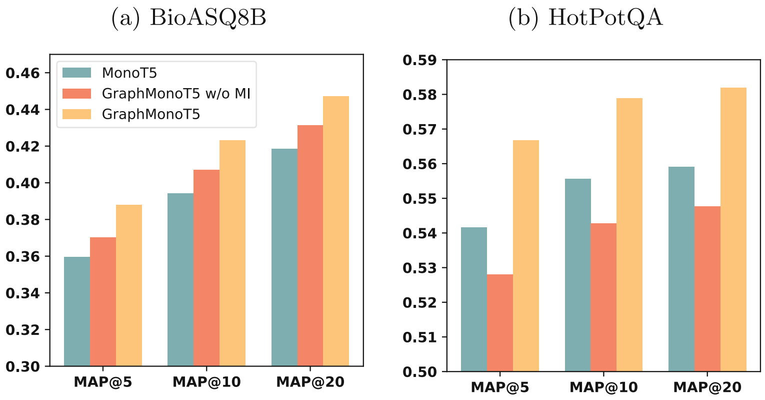


Fig. 1. Performance comparison of models in terms of MAP@k for BioASQ8B and HotPotQA test datasets.

without mutual information fusion drops on the HotpotQA dataset, which shows that effective language-graph fusion is required in an open-domain setup where the sub-graph of the question-document pair becomes sparse and comparatively larger (*cf.* Table 2) compared to the biomedical domain.

Qualitative Analysis: We have also performed qualitative analysis on the retrieved documents from MonoT5 and GraphMonoT5 models and observed that **(a)** MonoT5 model benefited from the world knowledge learned during pre-training stages and was able to retrieve the document where the query was syntactically and semantically aligned to the document, **(b)** however, MonoT5 model lacks the external biomedical knowledge and is not able to infer the underlying relations among the biomedical entities, leading to the incorrect document retrieval. In contrast, the proposed GraphMonoT5, which has learned the biomedical relations amongst the entities via UMLS and DrugBank, was able to retrieve the correct document. We have provided the query-documents examples comparing MonoT5 and GraphMonoT5 in Appendix D [1].

5 Conclusion

This work proposed an effective approach to re-rank the documents by utilizing the knowledge graphs and integrating them into the PLMs. To effectively fuse the language and graph information in the knowledge-enriched framework, we introduced a mutual information-based objective function, which ensures the fused representations are non-redundant and informative in nature. Extensive experiments on biomedical and open-domain datasets show the effectiveness of the approach.

Acknowledgments. This work was supported by the NLM Intramural Research Program and utilized the computational resources of the HPC Biowulf cluster (http://hpc.nih.gov).

References

1. Appendix (2024). https://drive.google.com/file/d/1-3SUGECi5x7bZwWCeP4jcRC52YogwA-I/view. Accessed 22 April 2024
2. Dai, Z., Callan, J.: Context-aware term weighting for first stage passage retrieval. In: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 1533–1536 (2020)
3. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers, pp. 4171–4186 (2019)
4. Hofstätter, S., Lin, S.C., Yang, J.H., Lin, J., Hanbury, A.: Efficiently teaching an effective dense retriever with balanced topic aware sampling. In: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 113–122 (2021)
5. Huang, L., Wu, L., Wang, L.: Knowledge graph-augmented abstractive summarization with semantic-driven cloze reward. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 5094–5107 (2020)
6. Karpukhin, V., et al.: Dense passage retrieval for open-domain question answering. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 6769–6781 (2020)
7. Kazaryan, A., Sazanovich, U., Belyaev, V.: Transformer-based open domain biomedical question answering at bioasq8 challenge. In: CLEF (Working Notes) (2020)
8. Khattab, O., Zaharia, M., Zaharia, M.: ColBERT: efficient and effective passage search via contextualized late interaction over BERT. In: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 39–48 (2020)
9. Kingma, D.P., Welling, M.: Auto-encoding variational Bayes. In: Bengio, Y., LeCun, Y. (eds.) 2nd International Conference on Learning Representations. ICLR 2014, Banff, AB, Canada, 14–16 April 2014, Conference Track Proceedings (2014). http://arxiv.org/abs/1312.6114
10. Liu, Z., Shen, Y., Lakshminarasimhan, V.B., Liang, P.P., Zadeh, A.B., Morency, L.P.: Efficient low-rank multimodal fusion with modality-specific factors. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 2247–2256 (2018)
11. Luo, M., Mitra, A., Gokhale, T., Baral, C.: Improving biomedical information retrieval with neural retrievers. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 36, no. 10, pp. 11038–11046 (2022). https://doi.org/10.1609/aaai.v36i10.21352, https://ojs.aaai.org/index.php/AAAI/article/view/21352
12. Nentidis, A., et al.: Overview of BioASQ 2020: the eighth BioASQ challenge on large-scale biomedical semantic indexing and question answering. In: Arampatzis, A., et al. (eds.) CLEF 2020. LNCS, vol. 12260, pp. 194–214. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58219-7_16

13. Nogueira, R., Jiang, Z., Pradeep, R., Lin, J.: Document ranking with a pretrained sequence-to-sequence model. In: Findings of the Association for Computational Linguistics: EMNLP 2020, pp. 708–718 (2020)
14. Pappas, D., Stavropoulos, P., Androutsopoulos, I.: AUEB-NLP at BioASQ 8: biomedical document and snippet retrieval. In: CLEF (2020)
15. Raffel, C., et al.: Exploring the limits of transfer learning with a unified text-to-text transformer. J. Mach. Learn. Res. **21**(140), 1–67 (2020)
16. Rahman, W., et al.: Integrating multimodal information in large pretrained transformers. In: Proceedings of the Conference. Association for Computational Linguistics. Meeting, vol. 2020, p. 2359. NIH Public Access (2020)
17. Robertson, S., Zaragoza, H., et al.: The probabilistic relevance framework: Bm25 and beyond. Found. Trends® Inf. Retrieval **3**(4), 333–389 (2009)
18. Thakur, N., Reimers, N., Rücklé, A., Srivastava, A., Gurevych, I.: Beir: a heterogeneous benchmark for zero-shot evaluation of information retrieval models. In: Thirty-Fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2) (2021)
19. Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., Bengio, Y.: Graph attention networks. In: International Conference on Learning Representations (2018)
20. Voorhees, E., et al.: TREC-COVID: constructing a pandemic information retrieval test collection. In: ACM SIGIR Forum. **54**, 1–12. ACM, New York (2021)
21. Xiong, L., et al.: Approximate nearest neighbor negative contrastive learning for dense text retrieval. In: International Conference on Learning Representations (2020)
22. Yang, Z., et al.: HotpotQA: a dataset for diverse, explainable multi-hop question answering. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pp. 2369–2380 (2018)
23. Yasunaga, M., Ren, H., Bosselut, A., Liang, P., Leskovec, J.: QA-GNN: reasoning with language models and knowledge graphs for question answering. In: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 535–546 (2021)
24. Zadeh, A., Chen, M., Poria, S., Cambria, E., Morency, L.P.: Tensor fusion network for multimodal sentiment analysis. In: Proceedings of the 2017 Conference on EMNLP, pp. 1103–1114 (2017)
25. Zhang, X., et al.: Greaselm: graph reasoning enhanced language models. In: International Conference on Learning Representations (2021)
26. Zhao, T., Lu, X., Lee, K.: Sparta: efficient open-domain question answering via sparse transformer matching retrieval. In: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 565–575 (2021)

Enhancing Abstract Screening Classification in Evidence-Based Medicine: Incorporating Domain Knowledge into Pre-trained Models

Regina Ofori-Boateng[1](✉), Magaly Aceves-Martins[2], Nirmalie Wirantuga[1], and Carlos Francisco Moreno-García[1]

[1] School of Computing, Robert Gordon University, Aberdeen, Scotland
{r.ofori-boateng,c.moreno-garcia}@rgu.ac.uk
[2] The Rowett Institute, University of Aberdeen, Aberdeen, Scotland

Abstract. Evidence-based medicine (EBM) represents a cornerstone in medical research, guiding policy and decision-making. However, the robust steps involved in EBM, particularly in the abstract screening stage, present significant challenges to researchers. Numerous attempts to automate this stage with pre-trained language models (PLMs) are often hindered by domain-specificity, particularly in EBMs involving animals and humans. Thus, this research introduces a state-of-the-art (SOTA) transfer learning approach to enhance abstract screening by incorporating domain knowledge into PLMs without altering their base weights. This is achieved by integrating small neural networks, referred to as knowledge layers, within the PLM architecture. These knowledge layers are trained on key domain knowledge pertinent to EBM, PICO entities, PubmedQA, and the BioASQ 7B biomedical Q&A benchmark datasets. Furthermore, the study explores a fusion method to combine these trained knowledge layers, thereby leveraging multiple domain knowledge sources. Evaluation of the proposed method on four highly imbalanced EBM abstract screening datasets demonstrates its effectiveness in accelerating the screening process and surpassing the performance of strong baseline PLMs.

Keywords: Pre-trained Language Models · Domain Integration · Transfer Learning · Evidence-Based Medicine · Abstract Text Classification

1 Introduction

Evidence-Based Medicine (EBM) presents the highest form of reliable evidence in shaping healthcare policies and decision-making [1]. Generally, the process involves (i) formulating a protocol, (ii) defining the research question using entity frameworks such as PICO[1] to encapsulate the inclusion and exclusion criteria,

[1] where PICO denotes Population, Intervention, Comparison, Outcome.

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 261–272, 2024.
https://doi.org/10.1007/978-3-031-66538-7_26

(iii) searching, (iv) screening abstracts, (v) extracting and analysing data from pertinent articles, and (vi) interpreting and publishing the findings. This process although structured is labour-intensive, further exacerbated by the daily increase in published articles. It is reported that the typical time frame for completing an EBM is approximately 15 months [1]. Thus, most EBMs become outdated before completion, needing major revisions.

Among all the stages in EBM, *abstract screening* has been reported to be the most challenging stage [2]. For example, research indicates that an experienced researcher typically spends 30–90 sec screening a single abstract, and estimated that 5,000 publications usually require 8–125 hrs [2]. Numerous methodologies for automating this stage have been proposed [2] ranging from traditional machine learning (ML) models to advanced PLMs, where they are fully fine-tuned (FFT) on EBM abstract datasets. However, most of these approaches are hindered by domain specificity, especially in highly imbalanced studies involving humans and animals [3]. Furthermore, PLMs comprise an extensive number of parameters; thus, in FFT, the parameters of the PLMs are updated whenever a new EBM dataset is introduced, resulting in increased computational costs and memory requirements. To tackle these issues, this paper investigates a SOTA method to integrate domain knowledge into PLMs for abstract screening tasks.[2]

2 Related Work

Many methods have been proposed for abstract screening, from traditional ML algorithms like Support Vector Machine (SVM) and Naive Bayes (NB) to SOTA PLMs. Timsina et al. [4] proposed using ULMS as a feature extraction technique and a softMax SVM classifier for abstract classification. Almeida et al. [5] also suggested the addition of MeSH and keywords to the abstracts for training a decision tree classifier. Similarly, Kontonatsios et al. [6] presented using MesH heading to train a neural network, and [7] proposed using Latent Dirichlet Allocation (LDA). With the rise of PLMs, medical domain knowledge PLMs such as SciBERT, PubMedBERT (PMBERT), BioBERT and CBERT (CBERT) have been proposed. For example, for this task, Hasny et al. [8] proposed using variants of BERT base models such as BERT-Meduim, SciBERT, BioBERT and CBERT. Ofori-Boateng et al. [2] also presented attention mechanisms with LSTM and Bi-LSTM. Moreno et al. [9] presented a zero-shot classification method for abstract screening. Similarly, [10] also proposed using GPT. Despite their advancements, these PLMs were originally trained on unstructured corpora, lacking the structured domain knowledge essential for biomedical tasks. As such, these PLMs treat biomedical concepts as conventional tokens, limiting their effectiveness [3].

2.1 Research Questions

We explore integrating essential domain knowledge into the models to address these issues. Specifically, we focus on incorporating PICO entities along with two

[2] For reproducibility, the source code and datasets are available on Github. https://github.com/reginaofori/EBM-Domain-Integration-PLMs.

biomedical Q&A datasets, PubMedQA[3] and BioASQ 7B[4]. PICO entities are fundamental in EBM, while PubMedQA and BioASQ 7B offer formats similar to the EMB abstract datasets (context/abstracts, question, and decision) as in Tables 3 and 4 in the Appendix. To this end, we ask the following **RQs**:

1. How can the diverse domain knowledge crucial for abstract screening tasks be integrated into a base PLM without adjusting model parameters? We insert small neural networks (knowledge layers) into the layers of a base PLM, SciBERT, using the principle of adapters [11] and train on the domain knowledge. Our choice of SciBERT is from a practical viewpoint as it was trained to cover a broad biomedical domain, thus advantageous for this task.
2. What is the effect of different configurations of the knowledge layers (where they are inserted) on the downstream task? We investigate and compare three configurations of inserted networks to analyse their influence on the downstream task.
3. Can adapter-based tuning perform better than SOTA FFT PLMs proposed for EBM abstracts? We empirically compare the performance of the trained knowledge layers with FFT SciBERT. Additionally, we examine the transferability and modularity of the method by inserting the already-trained networks into variants; CBERT, PMBERT, and BioBERT, adapter-tuning them, and comparing them against their FFT versions.

3 Methodology

3.1 Class Imbalance: Back Translation

EBM abstract classification struggles with class imbalance, where the number of excluded abstracts outweighs the included. Traditional methods have been proposed, such as cost-sensitive classifiers and data resampling [4]. However, this study proposes a SOTA data augmentation technique to address this issue called *Back-translation*. It involves translating the original text into another language and then back into the original language, generating a paraphrased version. Despite potential inaccuracies that may be introduced during re-translation, this method has demonstrated effectiveness in NLP tasks [8]. For this study, the Google Translate API[5] was utilised to translate English abstracts in the training dataset into seven different source languages (Spanish, French, German, Italian, Chinese (simplified), Chinese (traditional), and Irish), followed by re-translation back to English. Notably, back translation was applied only to the minority class (include). Further details on partitioning the downstream dataset for translation are provided in Sect. 4.

[3] https://pubmedqa.github.io/.

[4] http://participants-area.bioasq.org/datasets.

[5] https://translate.google.com/.

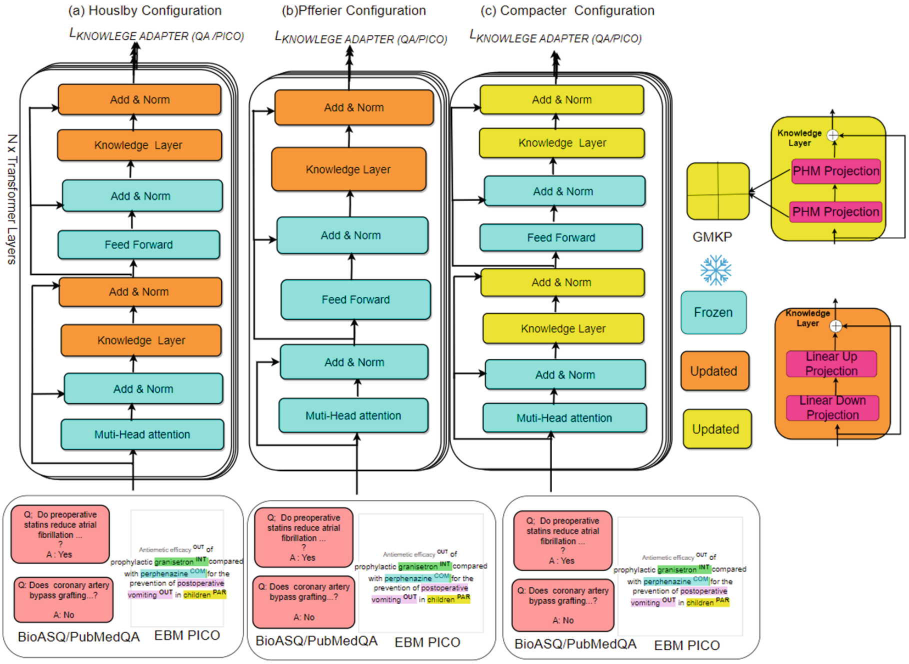


Fig. 1. Methodology for training the individual knowledge layers (PICO, BioASQ, and PubMedQA). The training involves 1) where the knowledge layer is inserted within the SciBERT PLM with frozen parameters and 2) where we investigate training with three configurations, in (a) the Houlsby (H), (b) the Pfeiffer, and in (c) the Compacter (C), a similar architecture of (a) but with a modification.

3.2 Overview of Adapters/Knowledge Layers

Adapters, originally proposed by Rebuffi et al. [13], are small trainable neural networks integrated within the layers of PLMs. An adapter consists of four main components: a FeedForward Linear Down Projection (FFD), FeedForward Linear Up Projection (FFU), a non-linear activation function (LeakyReLU), and a skip residual. The FFD and FFU reduce dimensionality, converting input from the PLM's high-dimensional space to a lower-dimensional one. For example, the FFD of the adapter maps the input data from the original high-dimensional space, d_{PLM}, to a much lower-dimensional space, h_{adapter}, where typically $h_{\text{adapter}} \ll d_{PLM}$. Readers are referred to [11] for the detailed mathematical explanation. The LeakyReLU enables the adapter to handle negative inputs, ensuring a more dynamic range for the activations. Lastly, the skip residual ensures the model doesn't lose essential information during transformation.

3.3 Approach–Training the Knowledge Layers

To address **RQ1** and **RQ2**, Fig. 1 illustrate the training of the three domain knowledge layers/adapters (PICO, PubMedQA, and BioASQ). We employ a comprehensive method in two phases: In Phase 1, the knowledge layers are integrated within every layer of the base SciBERT PLM to ensure a granular capture of information. In Phase 2, to examine the effect of different adapter configurations, we experiment with three distinct existing configurations in training the knowledge layers: (a) the Houlsby Configuration (H) [11], where the adapters modules are placed before the multi-head attention mechanism and the FeedForward layer of the SciBERT model as seen in Fig. 1, (b) The Pfeiffer Configuration (Pf) [15], where the adapter modules are placed exclusively after the FeedForward layer and (c) The Compacter Configuration (C) similar to the Houlsby configuration, but replaces the standard linear FFD and FFU with a more intricate Parameterised Hypercomplex Multiplication (PHM) layer [14]. The PHM layer uniquely determines its weights by computing the Global Multiplier of the Kronecker Product (GMKP) between two concise matrices. Readers are referred to the work done by [14] for a detailed explanation of how the GMKP and PHM work in the compacter. During the training of the knowledge layers, the integrated layers introduce trainable parameters, denoted by Φ_n which are only updated, while the core weights of the base SciBERT, Θ, remains static. This strategy accelerates the training process.

Training the Q&A Knowledge Layers–PubMedQA and BioASQ. The main goal of training a Q&A knowledge layer is to facilitate efficient transfer learning for our downstream abstract classification task, capitalising on the capabilities of SciBERT. PubMedQA and BioASQ, the two Q&A datasets used for training, are described in Table 4 in the appendix. In refining the training quality for PubMedQA (made up of three labels; yes/no/maybe), the“maybe” labels are excluded from both training and validation sets, ensuring a focus on clear-cut include (“yes”) or exclude (“no”) decisions to avoid potential ambiguities during training and in real life cases. Thus given the classification task (predicting “yes” or “no”), SciBERT is initialised with a binary sequence classification head, while the adapter module explained in Sect. 3.3 is introduced for training, keeping the main parameters of the SciBERT model frozen. The raw text Q&A data is tokenized using SciBERT’s tokenizer, combining questions with their corresponding contexts e.g., “[CLS] question [SEP] context [SEP] and the label “yes” is mapped to the label 1, while “no” is 0. Given our binary classification task, the cross-entropy loss function for optimisation is mathematically given as:

$$L_{\text{Q\&A}} = -\sum_{i=1}^{N} y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i) \tag{1}$$

where $L_{\text{Q\&A}}$ is the loss for the Q&A datasets, N is the total number of samples in each datasets, y_i denotes the actual label of the i-th sample, and $\hat{y}_i$ represents the predicted probability for the i-th sample being labelled as “yes”.

Table 1. Summary of results on the LN_19 (1) and AH_19(2) datasets.

DB		Methods	Precision	Recall	F1	WSS@95	AUC_PR	ROC
1	Know. Int. Baselines	KRISSBERT	0.9957	0.9957	0.9949	0.9465	0.82	0.97
		CODER-BERT	0.9948	0.9948	0.9935	0.9474	0.72	0.97
	FFT	SciBERT	0.9953	0.9957	0.9853	0.9448	0.79	0.95
	Modif. SciBERT	FPBPA(H)	**0.9957**	**0.9957**	**0.9949**	0.9465	**0.87** ↑ 0.05	**0.99**
		FPBPA(Pf)	0.9931	0.9931	0.9903	0.9491	0.85	0.87
		FPBPA (C)	0.9845	0.9922	0.9884	**0.9500**	**0.86**	**0.99**
	FFT	PMBERT	0.9940	0.9942	0.9920	0.9483	0.47	0.92
	Modif. PMBERT	FPBPA(H)	0.9948	0.9958	0.9935	0.9474	**0.81**	**0.99**
		FPBPA(Pf)	**0.9966** ↑ 0.09	**0.9970** ↑ 0.13	**0.9963** ↑ 0.14	**0.9591** ↑ 1.26	0.79	0.97
		FPBPA(C)	0.9942	0.9938	0.9935	0.9474	0.77	**0.99**
	FFT	BioBERT	0.9932	0.9934	0.9923	0.9483	0.57	0.98
	Modif. BioBERT	FPBPA(H)	0.9943	0.9940	0.9920	0.9483	0.66	**0.99**
		FPBPA(Pf)	**0.9952**	**0.9957**	**0.9949**	**0.9481**	**0.75**	**0.99**
		FPBPA(C)	0.9934	0.9931	0.9903	0.9442	0.67	0.98
	FFT	CBERT	0.9945	0.9948	0.9935	0.9474	0.62	0.98
	Modif. CBERT	FPBPA(H)	**0.9953**	**0.9957**	**0.9961**	0.9448	0.68	**0.99**
		FPBPA(Pf)	0.9939	0.9936	0.9934	0.9405	**0.77**	**0.99**
		FPBPA(C)	0.9931	0.9930	0.9903	**0.9491**	0.67	0.98
2	Know. Int. Baselines	KRISSBERT	0.9745	0.9777	0.9728	0.9388	0.46	0.82
		CODER-BERT	0.9428	0.9710	0.9567	0.9500	0.61	0.88
	FFT	SciBERT	0.9539	0.9725	0.9600	**0.9555** ↑ 0.55	0.69	0.93
	Modif. SciBERT	FPBPA(H)	0.9821	0.9821	0.9754	0.9299	**0.70** ↑ 0.24	**0.95** ↑ 0.13
		FPBPA(Pf)	**0.9832** ↑ 0.87	**0.9829** ↑ 0.52	**0.9794** ↑ 0.66	0.9433	0.52	0.92
		FPBPA(C)	0.9715	0.9762	0.9713	0.9388	0.57	0.85
	FFT	PMBERT	0.9682	0.9657	0.9675	0.9433	0.50	0.96
	Modif. PMBERT	FPBPA(H)	**0.9775**	**0.9769**	**0.9694**	0.9433	**0.58**	**0.91**
		FPBPA(Pf)	0.9650	0.9739	0.9633	**0.9500**	0.41	0.82
		FPBPA(C)	0.9719	0.9754	0.9691	0.9388	0.47	0.87
	FFT	BioBERT	0.9662	0.9721	0.9680	0.9188	0.60	0.88
	Modif. BioBERT	FPBPA(H)	**0.9728**	**0.9756**	**0.9695**	0.9433	**0.69**	**0.93**
		FPBPA(Pf)	0.9650	0.9739	0.9633	**0.9500**	0.41	0.82
		FPBPA(C)	0.9663	0.9747	0.9659	0.9433	0.57	0.86
	FFT	ClincalBERT	0.9439	0.9699	0.9519	0.9478	0.33	0.76
	Modif. CBERT	FPBPA(H)	**0.9532**	**0.9717**	**0.9584**	**0.9478**	**0.52**	**0.94**
		FPBPA(Pf)	0.9427	0.9688	0.9556	0.9478	0.34	0.95
		FPBPA(C)	0.9458	0.9677	0.9567	0.9411	0.20	0.82

Training EBM-PICO Knowledge Layer. The PICO framework is a fundamental structure for formulating clinical questions in EBM. In training the PICO adapter, the objective was to capture the inherent relationships embedded in PICO tags shown in Table 4 in the appendix. Thus, we implemented a token classification methodology, on the EBM PICO data tags. To mitigate training bias, the zero entity class was strategically excluded from the EBM-PICO dataset, creating an effective learning environment for the remaining relevant classes. Each relevant token is encoded with SciBERT into which the adapter module is integrated for training the EBM-PICO. After this encoding process, the token is directed to a classification layer to be classified into one of the three distinct PICO tags. The training objective for this is optimised using the cross-entropy loss function:

$$L_{\text{PICO}} = -\sum_{i=1}^{N} \left[y_i^{\text{PAR}} \log(\hat{y}_i^{\text{PAR}}) + y_i^{\text{INT}} \log(\hat{y}_i^{\text{INT}}) + y_i^{\text{OUT}} \log(\hat{y}_i^{\text{OUT}}) \right] \tag{2}$$

where $y_i^{\text{PAR}}, y_i^{\text{INT}}, y_i^{\text{OUT}}$, $\hat{y}_i^{\text{PAR}}, \hat{y}_i^{\text{INT}}, \hat{y}_i^{\text{OUT}}$ represent the ground-truth labels and the corresponding predicted probability distributions for the i i-th token, respectively, within the categories of Participants, Intervention/Comparison, and Outcome, and N encapsulates the cumulative count of tokens within the dataset.

3.4 Fusing the Trained Knowledge Layers/Adapters

To address **RQ3**, we integrate and tune the trained adapters in SciBERT, PMBERT, BioBERT and CBERT to show transferability. The individually trained adapters for PICO, PubMedQA, and BioASQ encapsulate different facets of information, each with its relevance to the downstream abstract task. Thus, to leverage the variability in the information stored by each trained adapter, we employ AdapterFusion [15]. AdapterFusion functions analogously to the attention layer in a standard transformer model, where the primary output from the PLM operates as the query. In contrast, the outputs from the various adapters act as keys and values. Readers are referred to the work done by [15] for further details. For clarity in this work, the combination of the trained PICO, PubMedQA and BioASQ is referred to as FPBPA.

4 Experimental Setup

Downstream SR Datasets for Evaluation. The proposed model was evaluated on four complex highly imbalanced EBM abstract datasets. One of these datasets, the Aceves-Martins_2022 dataset (AM_22) [16], is private focusing on oral health in children and nutritional disparities among prisoners. The remaining datasets; Appenzeller-Herzog_2019 (AH_19), Van-Dis_2019 (VD_20) and Leenars_2019 (LN_19) are publicly available on Github[6]. Each study's research question and inclusion/exclusion criteria were combined to form the "question" and the abstract was the "context". A summary of the datasets is provided in Table 5.

[6] https://github.com/asreview/synergy-dataset.

Table 2. Summary of results on the VD_20 (1) and AM_22 (2) datasets. Similar to, here ↑ denotes the % increment of the best results compared to the strongest baseline (CODER-BERT). The **Bold** values represent scenarios where the FPBPA method outperforms the FFT PLMs baselines within its category (SciBERT, BioBERT, PMBERT, CBERT) for the dataset. **Bold** also denotes the overall best value for each metric e.g. precision, recall, and F1 in each dataset (LN_19, AH_19). The ↑ denotes the % increment of the best results compared to the strongest baseline (KRISSBERT).)

DB		Methods	Precision	Recall	F1	WSS@95	AUC_PR	ROC
1	Know. Int. Baselines	KRISSBERT	0.9717	0.9785	0.9741	0.9417	0.29	0.91
		CODER-BERT	0.9792	0.9829	0.9743	0.9428	0.36	0.87
	FFT	SciBERT	0.9718	0.9735	0.9783	0.9456	0.26	0.62
	Modif. SciBERT	FPBPA(H)	0.9760	0.9813	0.9750	**0.9464**	**0.31**	0.8
		FPBPA(Pf)	**0.9787**	**0.9818**	**0.9786**	0.9302	**0.31**	0.85
		FPBPA(C)	0.9725	0.9702	0.9714	0.9467	**0.31**	**0.89**
	FFT	PMBERT	0.9670	0.9685	0.9677	0.9329	**0.35**	**0.95** ↑ 0.08
	Modif. PMBERT	FPBPA(H)	**0.9735**	**0.9779**	**0.9753**	0.9379	0.30	0.87
		FPBPA(Pf)	0.9716	0.9791	0.9739	**0.9434**	0.27	0.89
		FPBPA(C)	0.9695	0.9768	0.9725	0.9412	0.18	0.75
	FFT	BioBERT	0.9675	0.9618	0.9657	0.9461	0.35	0.76
	Modif. BioBERT	FPBPA(H)	0.9741	0.9807	0.9743	0.9461	0.24	0.76
		FPBPA(Pf)	**0.9809** ↑ 0.17	**0.9873** ↑ 0.44	**0.9772** ↑ 0.29	**0.9472** ↑ 0.44	0.25	0.77
		FPBPA(C)	0.9742	0.9807	0.9735	0.9461	**0.38** ↑ 0.02	**0.89**
	FFT	CBERT	0.9663	0.9624	0.9774	0.9445	0.29	0.71
	Modif. CBERT	FPBPA(H)	0.9771	0.9796	0.9782	0.9417	0.29	**0.87**
		FPBPA(Pf)	**0.9774**	**0.9818**	**0.9781**	**0.9447**	**0.30**	0.79
		FPBPA(C)	0.9707	0.9791	0.9732	0.9351	0.25	0.83
2	Know. Int. Baselines	KRISSBERT	0.9935	0.9939	0.9936	0.9393	0.75	0.98
		CODER-BERT	0.9953	0.9954	0.9953	0.9377	0.73	0.94
	FFT	SciBERT	0.9925	0.9931	0.9925	0.9210	0.64	0.89
	Modif. SciBERT	FPBPA(H)	0.9944	0.9946	0.9945	0.9370	0.77	0.97
		FPBPA(Pf)	**0.9956** ↑ 0.03	**0.9959** ↑ 0.05	**0.9959** ↑ 0.06	**0.9485** ↑ 1.08	**0.87** ↑ 1.14	**0.99** ↑ 0.01
		FPBPA(C)	0.9906	0.9916	0.9903	0.9385	0.73	0.92
	FFT	PMBERT	0.9905	0.9904	0.9877	**0.9466**	**0.86**	**0.99**
	Modif. PMBERT	FPBPA(H)	**0.9938**	**0.9935**	**0.9936**	0.9358	**0.85**	**0.99**
		FPBPA(Pf)	0.9928	0.9927	0.9928	0.9366	0.73	0.98
		FPBPA(C)	0.9920	0.9923	0.9910	0.9439	0.73	0.98
	FFT	BioBERT	0.9920	0.9927	0.9919	0.9420	0.80	**0.99**
	Modif. BioBERT	FPBPA(H)	0.9930	0.9935	0.9931	0.9397	0.64	0.92
		FPBPA(Pf)	**0.9944**	**0.9946**	**0.9945**	**0.9431**	0.72	0.96
		FPBPA (C)	0.9933	0.9931	0.9932	0.9362	**0.84**	**0.99**
	FFT	ClincalBERT	0.9941	0.9927	0.9932	0.9328	0.83	**0.99**
	Modif. CBERT	FPBPA(H)	0.9921	0.9925	0.9923	0.9397	0.82	**0.99**
		FPBPA(Pf)	**0.9948**	**0.9950**	**0.9948**	**0.9404**	**0.84**	0.98
		FPBPA (C)	0.9939	0.9943	0.9938	0.9389	0.78	0.93

Implementation and Hyperparameters. The AdapterHub[7], HuggingFace library[8], and the PyTorch framework were employed for training the knowledge adapters evaluation. Our experimental setup was done with Nvidia 2080Ti GPUs. To ensure uniform input dimensions during the training of the knowledge layers, we truncated/pad sequences to a consistent length of 512 tokens. We split each of the datasets in Table 4 into 90% train and 10% validation split, to find the optimal hyperparameters. In training the PICO adapter, we deployed the following hyper-parameters; warmup step: $[0, \mathbf{500}, 1000]$, epochs: $[3, \mathbf{5}, 10, 20]$, batch size: $[8, \mathbf{16}, 64, 256]$, weight decay: $[\mathbf{0.0}, 0.1, 0.01, 0.001]$ and learning rate: $[\mathbf{1}e^{-4}, 3e^{-5}, 1e^{-5}]$ with the AdamW as the optimizer. Similarly, the same hyperparameters were in training in PubMedQA and BioASQ adapters. However, the best-performing batch size and epochs for the PubMedQA were **64** and **3**, whereas the best-performing learning rate for the BioASQ was $3e^{-5}$. We modulated three random seeds (42, 10 and 50) and reported on the aggregated results over the iterations to ensure robustness.

Evaluation Metrics and Baselines. We report on the weighted average: precision and recall, AUC Precision-recall, AUC ROC and Work saved oversampling (WSS@95%) [5] which measures how much human burden the model can reduce. During the evaluation, we split the downstream dataset into a 60/40 train test set. We applied the back translation augmentation technique described in Sect. 3.1 only to the minority (include) in the train set. Further, we partitioned the final augmented and initial train sent into a 10% dev set whilst we reported the average runs on the unaugmented test set. To compare the performance of our method, we explore existing FFT proposed for abstract screening tasks. As such, **FFT-PMBERT**, **FFT-SciBERT**, **FFT-BioBERT** and **FFT-CBERT**. To further validate the performance of our model, we compare with two SOTA knowledge integrated PLMs **CODER-BERT**[9], a UMLS triples embedding integration via contrastive learning and **KRISSBERT**[10], a PMBERT that utilises self-supervised learning for entity linking.

5 Results and Discussion

Tables 1 and 2 show the results obtained from evaluating the adapter-based tuning against the FFT biomedical variants PLMs and existing strong knowledge PLM integrated baselines (CODER-BERT and KRISSBERT). Generally, the tables demonstrate a consistent trend across various PLMs: tuning FPBPA (H, Pf, or C) within the PLMs leads to notable metric improvements compared to the baseline. This finding addresses **RQ3** indicating the effectiveness of FPBPA for the EBM abstract screening task. Further discussion is as follows:

[7] https://adapterhub.ml/.
[8] https://huggingface.co/docs/transformers/index.
[9] https://huggingface.co/GanjinZero/UMLSBert_ENG.
[10] https://huggingface.co/microsoft/BiomedNLP-KRISSBERT-PubMed-UMLS-EL.

Can Adapter-Based Tuning Perform Better than SOTA FFT PLMs? Discussing Table 1 for the highly imbalanced ratio (IR) dataset LN_19 (IR 1:341), FPBPA(Pf) consistently achieves high precision, recall, WSS@95, and F1 score compared to the baselines, particularly in PMBERT. Additionally, FPBPA(H) and FPBPA(C) also show competitive performance, especially in terms of precision and AUC_PR. Similarly, for AH_19 (IR 1:98), SciBERT-FPBPA(Pf) consistently outperforms the strongest baseline and FFT PLMs.
In Table 2, for the VD_20 (1: 126) dataset, BioBERT FPBPA(Pf) achieves higher precision, recall, WSS@95 and F1 score compared to the strongest baseline and FFT PLMs. Similarly, for AM_22 (1:188), SciBERT-FPBPA (Pf) outperforms the FFT and the strongest knowledge-integrated baseline across all metrics.

What is the Effect of the Different Configurations of Knowledge Layers? The different FPBPA configurations (H, Pf, C) exhibit variable impacts on different datasets seen in Tables 1 and 2. To summarise the analysis, the FPBPA(Pf) shows strength in the extremely imbalanced datasets compared to the H and C. Thus, in practicality, the use of FPBPA(Pf) may be useful in situations where the EBM to be done is broad and may lead to broad search strings, hence encompassing lots of irrelevant literature compared to the number of relevant as in the case of **LN_19** and **AM_22** dataset.

6 Conclusion and Future Works

This research explores a SOTA transfer learning method that infuses domain-specific insights into PLMs using adapters. Utilizing the PICO framework alongside resources like PubMedQA and BioASQ Q&A, our technique improves PLM capabilities for EBM abstract screening, which is critical for enhancing clinical decisions and policies. Through detailed experiments, we demonstrate that our method delivers promising outcomes across various metrics, including precision, recall, F1 score, and WSS@95. Looking ahead, we plan to incorporate additional domain-specific resources such as UMLS, DisGeNET, and the UNIPROT knowledge database to broaden our approach's relevance. Currently, our research centres on the BERT model, but future investigations will include other SOTA PLMs like GPT and LLaMA. Furthermore, a future work will be to conduct a comparative analysis of our method against baseline models such as SVM and NB +/- UMLS, employing keyword search techniques like cTAKES or a MetaMap-based model using TF-IDF or n-gram analysis.

A Appendix

Table 3. Format of the EBM abstract screening dataset

Abstract (Abs)	**Research Question (RQ)**	**Decision**
1. Glycosylated haemoglobins and weights were recorded for 200 consecutive diabetic...	What is the prevalence of overweight and obesity among imprisoned populations worldwide?	Exclude
2. Childhood dental caries and obesity are prevalent health problems. Results from previous studies of the caries-obesity	Is there an association between obesity or overweight and poor oral health among Mexican children and adolescents?	Include

Table 4. Statistics of datasets used to train the knowledge layers/adapters

Dataset	**Adapter**	**Format**	**Size**
EBM-PICO[a]	PICO	(I-INT, I-OUT, I-PAR)[b]	5000
PubMedQA	P-QA	(Context/Question/labels(yes/no/maybe))	211.3K
BioASQ	B-ASQ	(Context/Question/labels(yes/no))	6676

[a] https://github.com/bepnye/EBM-NLP

[b] where Participants is (I-PAR), Outcome (I-OUT), and a combination of Intervention/Comparison as (I-INT)

Table 5. Summary of the datasets ranging from human to animal study, where IR = Imbalance Ratio, the variables used for each EBM dataset are in Table 3.

Name_of_dataset	**Subject**	**Total_papers**	**Relevant**	**Irrelevant**	**IR**	**Abs Len (Avg)**
Aceves-Martins_2022(AM_22)	Nutritional status of prisoners	13022	69	12953	1:188	1765.37
Appenzeller-Herzog_2019(AH_19)	Therapy for Wilson Disease	2873	29	2844	1:98	1282.35
Leenars_2019(LS_19)	Animal to human translation	5812	17	5795	1:341	1458.40
Van_Dis_2020(VD_20)	Cognitive Behavioral Therapy	9128	72	9056	1:126	1473.08

References

1. Burns, P.B., Rohrich, R.J., Chung, K.C.: The levels of evidence and their role in evidence-based medicine. Plast. Reconstr. Surg. **128**(1), 305–310 (2011)
2. Ofori-Boateng, R., Aceves-Martins, M., Jayne, C., Wiratunga, N., Moreno-Garcia, C.F.: Evaluation of attention-based LSTM and Bi-LSTM networks for abstract text classification in systematic literature review automation. Procedia Comput. Sci. **222**, 114–126 (2023)
3. Xie, Q., Bishop, J.A., Tiwari, P., Ananiadou, S.: Pre-trained language models with domain knowledge for biomedical extractive summarization. Knowl. Based Syst. **252**, 109460 (2022)

4. Timsina, P., Liu, J., El-Gayar, O.: Advanced analytics for the automation of medical systematic reviews. Inf. Syst. Front. **18**(2), 237–252 (2015)
5. Almeida, H., Meurs, M.-J., Kosseim, L., Tsang, A.: Data sampling and supervised learning for HIV literature screening. IEEE Trans. Nanobiosci. **15**(4), 354–361 (2016)
6. Kontonatsios, G., Spencer, S., Matthew, P., Korkontzelos, I.: Using a neural network-based feature extraction method to facilitate citation screening for systematic reviews. Expert Syst. Appl. X **6**, 100030 (2020)
7. Natukunda, A., Muchene, L.K.: Unsupervised title and abstract screening for systematic review: a retrospective case-study using topic modelling methodology. Syst. Rev. **12**(1), 1 (2023)
8. Hasny, M., et al.: BERT for complex systematic review screening to support the future of medical research. In: Artificial Intelligence in Medicine (2023)
9. Moreno-Garcia, C.F., Jayne, C., Elyan, E., Aceves-Martins, M.: A novel application of machine learning and zero-shot classification methods for automated abstract screening in systematic reviews. Decis. Analytics J. **6**, 100162 (2023)
10. Guo, E., Gupta, M., Deng, J., Park, Y.-J., Paget, M., Naugler, C.: Automated Paper Screening for Clinical Reviews Using Large Language Models (2023)
11. Houlsby, N., et al.: Parameter-Efficient Transfer Learning for NLP (2019)
12. Maloof, M.A.: Learning when data sets are imbalanced and when costs are unequal and unknown. In: ICML-2003 Workshop on Learning from Imbalanced Data Sets II (2003)
13. Rebuffi, S.-A., Bilen, H., Vedaldi, A.: Learning multiple visual domains with residual adapters. In: Guyon, I., Von Luxburg, U., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) Advances in Neural Information Processing Systems, vol. 30 (2017)
14. Mahabadi, R.K., Henderson, J., Ruder, S.: Compacter: Efficient Low-Rank Hypercomplex Adapter Layers (2021)
15. Pfeiffer, J., Kamath, A., Rücklé, A., Cho, K., Gurevych, I.: AdapterFusion: non-destructive task composition for transfer learning. In: Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume (2021)
16. Aceves-Martins, M., et al.: Interventions to treat obesity in Mexican children and adolescents: systematic review and meta-analysis. Nutr. Rev. **80**(3), 544–560 (2022)

Exploring Pre-trained Language Models for Vocabulary Alignment in the UMLS

Xubing Hao[1], Rashmie Abeysinghe[2], Jay Shi[3], and Licong Cui[1](✉)

[1] McWilliams School of Biomedical Informatics, The University of Texas Health Science Center at Houston, Houston, TX 77030, USA
{xubing.hao,licong.cui}@uth.tmc.edu

[2] Department of Neurology, The University of Texas Health Science Center at Houston, Houston, TX 77030, USA
rashmie.abeysinghe@uth.tmc.edu

[3] Intermountain Healthcare, Denver, CO 80206, USA

Abstract. The Unified Medical Language System (UMLS) Metathesaurus integrates and aligns terms from hundreds of biomedical vocabularies. In this paper, we investigate the efficacy of Pre-trained Language Models (PLMs) for vocabulary alignment in the UMLS Metathesaurus. We frame the problem as two Natural Language Processing tasks: Text Classification and Text Generation. We fine-tune four opensource cutting-edge PLMs including BERT and RoBERTa, GPT-2, and BLOOM. Experiments show that the best model is RoBERTa achieving a precision, recall, and F1 score of 0.965, 0.940, and 0.952 respectively. In addition, incorporation of contextual information in the inputs improves the model performance in the Text Classification task, albeit with a limited impact on the Text Generation task. Domain expert evaluation of 100 randomly selected instances generated by the best model revealed that 78 of them as valid synonymous terms, indicating the promise of PLMs in enhancing the mapping quality of the UMLS Metathesaurus.

Keywords: UMLS Metathesaurus · Pre-trained Language Models · Vocabulary Alignment

1 Introduction

The Unified Medical Language System (UMLS) Metathesaurus is a biomedical vocabulary integration system created by the US National Library of Medicine [2]. It integrates terms from hundreds of biomedical vocabularies including SNOMED CT, National Cancer Institute (NCI) Thesaurus, and Current Procedural Terminology (CPT). The basic building block of the UMLS Metathesaurus is the atom, a term from a specific source vocabulary that is allocated a distinct Atom Unique Identifier (AUI) [2]. A concept denotes a particular meaning aggregating all the atoms from any vocabulary that convey this particular meaning in any form and is designated with a Concept Unique Identifier (CUI). For instance, five atoms are grouped under the UMLS concept "Diabetic

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 273–278, 2024.
https://doi.org/10.1007/978-3-031-66538-7_27

Nephropathy" with a CUI of C0011881, which are "Diabetic Nephropathy" from NCI Thesaurus (AUI A17683370), "Nephropathy, Diabetic" from MeSH (AUI A0438822), "Disorder of kidney due to diabetes mellitus" from SNOMED CT (US Edition) (AUI A31540124), "Diabetic Kidney Disease" from LOINC (AUI A28306370), and "Diabetes with renal manifestations" from ICD-9-CM (AUI A8340913).

Given that the UMLS Metathesaurus incorporates millions of terms from more than 180 source vocabularies, integrating and aligning these terms is challenging. Constructing and maintaining the UMLS Metathesaurus entails lexical and semantic techniques to identify potential synonyms that are further reviewed and finalized by human reviewers which is time-consuming and labor intensive [4]. To address these issues, researchers have studied automated techniques such as rule-based methods [6], Siamese Networks using Long Short-Term Memory (LSTM) [1], knowledge graph embedding-based approaches [5], Bidirectional Encoder Representations from Transformers (BERT) models [7], and Graph Convolutional Neural Networks (GCN) [3] for aligning terms within the UMLS Metathesaurus.

In this work, we explore the potential of four open-source Pre-trained Language Models (PLMs) for facilitating vocabulary alignment within the UMLS Metathesaurus by framing the research question into Natural Language Processing (NLP) tasks: Text Classification and Text Generation. We fine-tune BERT and RoBERTa for the Text Classification task, and GPT-2 and BLOOM for the Text Generation task. Our research is structured around the following research questions: (RQ1) How do PLMs perform on synonymy identification in a Text Classification setting; (RQ2) How do PLMs perform on synonymy identification in a Text Generation setting; and (RQ3) How effective are PLMs in identifying potentially missing synonymous terms in the UMLS Metathesaurus?

2 Methods

2.1 Dataset Construction

Two atoms *A* and *B* grouped under the same UMLS concept but originating from two different source vocabularies will form a synonymous atom pair *(A, B)*. For example, "Diabetic Nephropathy" in NCI Thesaurus and "Nephropathy, Diabetic" in MeSH will serve as a synonymous atom pair. For each synonymous atom pair *(A, B)*, we replace *B* with *A*'s most lexically similar atom *X* (according to cosine similarity score) that does not originate from the same vocabulary as *A* to form a non-synonymous atom pair *(A, X)*. For instance, "Diabetic Nephropathy" in NCI Thesaurus and its lexically similar atom "Diabetic nephropathy screening" in SNOMED CT form a non-synonymous atom pair. We group synonymous atom pairs and their corresponding non-synonymous atom pairs according to the UMLS concepts from which the synonymous atom pairs were generated and split these groups into training, validation, and testing sets with a ratio of 8:1:1, ensuring that two synonymous atom pairs generated from one concept do not spread across training/validation/testing sets.

2.2 Experiment Setup

In RQ1, we explore the efficacy of PLMs in synonymy identification, framing it as a Text Classification task with two classes: (1) positive class, consisting of pairs of terms that are synonymous; and (2) negative class, comprising pairs of terms that are not synonymous. We fine-tune BERT and RoBERTa pre-trained models for this task.

In RQ2, we approach synonymy identification as a Text Generation task. In this context, we provide the GPT-2 and BLOOM models with instructions and information about the atoms, requesting the models to generate a response regarding whether the two atoms are synonymous.

In RQ3, we delve into the proficiency of PLMs in detecting missing synonymous atoms within the UMLS Metathesaurus. Specifically, our objective is to uncover synonymous terms that have not been categorized under the same concept in the UMLS Metathesaurus. Within our testing set, when a pair of atoms is originally non-synonymous (a negative instance), but our model predicts it as synonymous, we consider these terms as likely candidates for missing synonymous terms.

2.3 Input and Prompt Design

Table 1 shows the design of our inputs and prompts tailored for various tasks. For the Text Classification task, we have devised two distinct input configurations: I_1 and I_2. Input I_1 is structured to include merely the names of the two atoms, which are delineated by a " |" symbol. Input I_2 is more comprehensive additionally including information about the source vocabularies and parent terms. For the Text Generation task, we have developed two distinct prompt configurations: P_1 and P_2. Each prompt for this task is structured into three segments: a task instruction, an input, and a response. The task instruction explicitly outlines the task to be performed. Prompt P_1 comprises solely the names of the two atoms while prompt P_2 provides additional information about their source vocabularies and parent terms.

Table 1. Input/Prompt design for different tasks.

Task	Input/Prompt Configuration	Design
Text Classification	Input I_1	{atom 1} | {atom 2}
Text Classification	Input I_2	{atom 1}. This term is from {atom 1's source terminology}. It is a subtype of {atom 1's parents}. | {atom 2}. This term is from {atom 2's source terminology} It is a subtype of {atom 2's parents}

(*continued*)

Table 1. (*continued*)

Task	Input/Prompt Configuration	Design
Text Generation	Prompt P_1	### Instruction: Classify if the two following terms are synonymous or not ### Input: Term 1: {atom 1} Term 2: {atom 2} ### Response:
Text Generation	Prompt P_2	### Instruction: Classify if the two following terms are synonymous or not ### Input: Term 1: {atom 1} This term is from {atom 1's source terminology}. It is a subtype of {atom 1's parents} Term 2: {atom 2} This term is from {atom 2's source terminology}. It is a subtype of {atom 2's parents} ### Response:

2.4 Evaluation

We present the performance metrics including precision, recall, and F1 score of the PLMs on the validation set. From the suggestions by the model exhibiting the highest F1 score on the testing set, we extract a randomly chosen subset of suggested potentially missing synonymous atoms for further evaluation by a domain expert with experience in clinical terminology assessment.

3 Results

In this study, we utilized the 2022 AA full version of the UMLS Metathesaurus with 16 million atoms grouped under 4 million UMLS concepts. Our constructed dataset comprised 17,710,981 synonymous atom pairs and 17,162,449 non-synonymous atom pairs. Further splitting resulted in 27,962,212; 3,414,455; and 3,496,793 in the training, validation, and testing sets respectively.

The models were trained on four NVIDIA A100-SXM4 graphics cards, each with 80GB of RAM. The hyperparameters employed during model training include a learning rate of 5e-5, a batch size of 256, a training epoch of 5, and AdamW optimizer.

The performance of the models in different settings is shown in Table 2. As can be seen, RoBERTa with input I_2 achieved the best performance with an F-1 score of 0.952 closely followed by BERT with the same input.

Utilizing the best model on the testing set, the model identified 67,150 atom pairs as potentially missing synonymous atom pairs. Manual evaluation of randomly selected

Table 2. Model performance in the Text Classification and Text Generation Settings.

Model	Task	Config	Precision	Recall	F1 score
BERT	Text Classification	I_1	0.949	0.923	0.936
BERT	Text Classification	I_2	0.970	0.933	0.951
RoBERTa	Text Classification	I_1	0.952	0.932	0.942
RoBERTa	Text Classification	I_2	0.965	0.940	**0.952**
GPT-2	Text Generation	P_1	0.882	0.884	0.883
GPT-2	Text Generation	P_2	0.874	0.906	0.890
BLOOM	Text Generation	P_1	0.929	0.925	0.927
BLOOM	Text Generation	P_2	0.918	0.889	0.903

100 atom pairs verified 78 as valid synonymous pairs. Table 3 lists five instances of these valid missing synonymous atom pairs.

Table 3. Five missing synonymous atoms validated by the domain expert.

Atom 1 (source vocabulary)	Atom 2 (source vocabulary)
Vinorelbine (as vinorelbine tartrate) 10 mg/mL solution for infusion (US Edition of SNOMED CT)	vinorelbine (as vinorelbine tartrate) 10 MG per 1 ML Injection (RxNorm)
Technetium Tc-99m albumin colloid (DrugBank)	technetium Tc 99m human serum albumin colloid (Physician Data Query)
hemolytic disease of the newborn (Consumer Health Vocabulary)	ABO; hemolytic disease (ICPC2 - ICD10 Thesaurus)
Enteropathy associated T-cell lymphoma (US Edition of SNOMED CT)	Lymphoma, T-Cell, Enteropathy-Associated (MeSH)
Acute renal failure with tubular necrosis (International Classification of Diseases)	acute renal failure due to tubular necrosis (diagnosis) (MEDCIN)

4 Discussion

Recent studies have applied deep learning techniques to vocabulary alignment in the UMLS. For instance, Nguyen et al. used a Siamese LSTM architecture for supervised learning [5]. Wijesiriwardene et al. adapted BERT-based models like BioBERT and SapBERT [7]. Our model surpasses these approaches with an F1 score of 0.952 compared to their 0.937 and 0.942 respectively, underscoring the effectiveness of PLMs in UMLS vocabulary alignment. In the future, we expect to further expand the comparisons to zero-shot and few-shot learning strategies using generative models. In addition, larger models like LLaMA2 70B and newer GPT models like GPT-3.5 and GPT-4 could also be investigated.

5 Conclusion

In this study, we evaluated the efficacy of PLMs including BERT, RoBERTa, GPT-2, and BLOOM for UMLS vocabulary alignment. Results showed that PLMs have strong potential in this domain, with the best model RoBERTa achieving a precision, recall, and F1 score of 0.965, 0.940, and 0.952 respectively. A manual evaluation revealed 78 out of 100 random predictions were valid missing synonyms indicating the promise of PLMs to facilitate vocabulary alignment in the UMLS.

Acknowledgments. This work was supported by the National Science Foundation (NSF) through grant 2047001 and National Institutes of Health (NIH) through grant R01NS116287. The content is solely the responsibility of the authors and does not necessarily represent the official views of the NSF or NIH.

Disclosure of Interests. The authors have no competing interests.

References

1. Bajaj, G., et al.: Evaluating deep learning models for vocabulary alignment at scale in the umls metathesaurus (2022)
2. Bodenreider, O.: The unified medical language system (umls): integrating biomedical terminology. Nucleic Acids Res. **32**(suppl 1), D267–D270 (2004)
3. Hao, X., Abeysinghe, R., Shi, J., Cui, L.: A GCN-based approach to uncover misaligned synonymous terms in the UMLS metathesaurus. In: AMIA Annual Symposium Proceedings, vol. 2023, p. 977. American Medical Informatics Association (2023)
4. Nguyen, V., Bodenreider, O.: UVA resources for the biomedical vocabulary alignment at scale in the UMLS metathesaurus. arXiv preprint arXiv:2205.10575 (2022)
5. Nguyen, V., et al.: Context-enriched learning models for aligning biomedical vocabularies at scale in the umls metathesaurus. In: Proceedings of the ACM Web Conference 2022, pp. 1037–1046 (2022)
6. Nguyen, V., Yip, H.Y., Bodenreider, O.: Biomedical vocabulary alignment at scale in the umls metathesaurus. In: Proceedings of the Web Conference 2021, pp. 2672– 2683 (2021)
7. Wijesiriwardene, T., et al.: UBERT: a novel language model for synonymy prediction at scale in the UMLS metathesaurus. arXiv preprint arXiv:2204.12716 (2022)

ICU Bloodstream Infection Prediction: A Transformer-Based Approach for EHR Analysis

Ortal Hirszowicz[1] and Dvir Aran[1,2](✉)

[1] Taub Faculty of Computer Science, Technion-Israel Institute of Technology, Haifa, Israel
dviraran@technion.ac.il

[2] Faculty of Biology, Technion-Israel Institute of Technology, Haifa, Israel

Abstract. We introduce RatchetEHR, a novel transformer-based framework designed for the predictive analysis of electronic health records (EHR) data in intensive care unit (ICU) settings, with a specific focus on bloodstream infection (BSI) prediction. Leveraging the MIMIC-IV dataset, RatchetEHR demonstrates superior predictive performance compared to other methods, including RNN, LSTM, and XGBoost, particularly due to its advanced handling of sequential and temporal EHR data. A key innovation in RatchetEHR is the integration of the Graph Convolutional Transformer (GCT) component, which significantly enhances the ability to identify hidden structural relationships within EHR data, resulting in more accurate clinical predictions. Through SHAP value analysis, we provide insights into influential features for BSI prediction. RatchetEHR integrates multiple advancements in deep learning which together provide accurate predictions even with a relatively small sample size and highly imbalanced dataset. This study contributes to medical informatics by showcasing the application of advanced AI techniques in healthcare and sets a foundation for further research to optimize these capabilities in EHR data analysis.

Keywords: Transformer · Electronic health records · Blood stream infection

1 Introduction

The advent of Electronic Health Records (EHRs) has revolutionized the landscape of healthcare data management, offering unprecedented opportunities for enhancing patient care and clinical decision-making. Particularly in Intensive Care Units (ICUs), where patients are at high risk and require close monitoring, the effective analysis of EHRs can be a lifesaver. One of the most critical applications of EHR analysis in ICU settings is the early detection of bloodstream infections (BSIs), a condition associated with high morbidity and mortality rates [1]. Traditional approaches to predicting BSIs have relied on a variety of statistical and machine-learning approaches, yet these have often fallen short due to the complex, temporal, and high-dimensional nature of EHR data.

Existing models for BSI prediction [2–5] have faced significant challenges in capturing the nuanced temporal dynamics and intricate feature inter-dependencies present in

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 279–292, 2024.
https://doi.org/10.1007/978-3-031-66538-7_28

EHR data. These limitations stem primarily from the challenges in processing sequential data effectively, leading to a loss of critical information. Furthermore, the high dimensionality and sparsity of EHR data, coupled with issues like class imbalance and the need for extensive data preprocessing, have further complicated the predictive analysis.

In response to these challenges, we introduce RatchetEHR, leveraging the power of transformer-based architecture [6], to analyze ICU EHRs for BSI prediction. This approach allows to effectively capture the sequential nature and hidden structures within the data. We show that this approach not only enhances prediction accuracy but also offers a deeper understanding of the underlying patterns and relationships in EHR data.

2 Related Work

The transformer architecture [6] emerges as a particularly fitting model for modeling EHR data due to several intrinsic properties of EHR and the strengths of transformers. EHR data is inherently complex, comprising long sequences of patient visits, each containing various medical elements like diagnoses, treatments, and observations. This complexity and sequential nature of EHR data align well with the capabilities of transformers. Transformers excel in handling sequential data, a property leveraged extensively in natural language processing (NLP). They are adept at capturing long-term dependencies and intricate relationships within sequences, which is crucial for interpreting EHR data where past medical events can significantly influence future health outcomes. The transformer's self-attention mechanism allows it to weigh the importance of different parts of the sequence differently. This aspect is particularly beneficial in EHR data, where not all medical events have equal relevance to a patient's current health status or future medical predictions. By focusing on more significant events in a patient's medical history, transformers can provide more accurate and personalized health predictions. Our project was mainly inspired by the following studies that used the Transformer model with electronic health record data:

GCT model. Prior research in EHR data representation primarily utilized the Bag of Words (BOW) approach, treating each medical feature as an isolated entity. This methodology, however, led to significant information loss about the physician's decision-making process. The Graph Convolutional Transformer (GCT) model, presented by Choi et al.[7], offers a robust solution to this issue by employing the Transformer architecture's self-attention mechanism. This mechanism effectively learns a hidden graphical structure, delineating the relationship between different EHR features, thereby overcoming the limitations of the BOW approach.

GCT represents EHR data as a two-dimensional matrix, where each cell indicates a connection between two features in the graph. This model's primary objective is to learn this hidden structure and utilize it for various predictive tasks, especially when explicit structural information is not available. The article highlighted that the learned hidden structure through GCT creates new embeddings, significantly enhancing the model's performance across various tasks.

However, GCT has certain limitations. Its analysis is confined to single-time intervals—specifically, individual hospital visits—potentially overlooking the continuum of patient care. Furthermore, by dividing features into three categories—diagnosis, treatment, and lab tests—GCT's generality across diverse medical scenarios is somewhat constrained. Despite these drawbacks, GCT's approach in learning and utilizing hidden EHR structures demonstrates significant improvements in model performance for tasks like readmission and mortality prediction, marking a notable advancement in the application of Transformer models in the realm of EHR data.

SARD model. Kodialam et al. [8] introduced a Transformer-based architecture known as SARD, innovatively combining embeddings for hospital visits, temporal embeddings, and a self-attention mechanism. This design deviates from the traditional positional embeddings, accommodating the non-uniform timing of hospital visits. In their study, they utilized a large dataset of administrative claims to predict end-of-life and surgeries in the next six months. We recently developed an extension of the SARD architecture, which we named STRAFE [9], with the goal of predicting time-to-event instead of fixed-time prediction and applied it to predict deterioration in chronic kidney disease.

One notable limitation of using claims data, and by extension the SARD model, is its exclusion of granular data from individual hospital visits, such as real-time monitoring signals (e.g., respiratory rate values). This gap highlights a potential area for model improvement in capturing finer details of patient care.

A unique aspect of the SARD model is its ability to discern connections between individual hospital visits, revealing a 'hidden structure' in the healthcare journey of a patient. However, it faces challenges in accurately representing scenarios with rapidly changing features, such as during an ICU stay. In such cases, the model may not fully capture the dynamic nature of a patient's condition, where variables like respiratory rate can fluctuate significantly over short periods.

3 Framework Overview

This RatchetEHR framework is primarily inspired by the SARD framework, which is used for claims data analysis. However, EHR data presents unique challenges compared to claims data, necessitating several key modifications in our approach.

First, unlike claims data which primarily consist of categorical features such as diagnosis and procedure codes, EHR data encompasses a wide array of numerical features. These include vital statistics like blood counts or respiratory rates, which are more akin to continuous signals than discrete categories.

Second, the temporal scope of the data differs markedly between these two domains. Claims data often span multiple visits over extended timeframes, offering a longitudinal view of a patient's health history. In contrast, EHR data, particularly in the context of ICU stays, tends to be more focused, typically concentrating on a single hospital admission.

In light of these differences, we made several adjustments to the original SARD framework to better suit the specific requirements of ICU EHR data analysis. RatchetEHR introduces a transformer-based architecture that is specifically designed for the task of analyzing ICU EHR data. Central to our framework is the ability to process ICU stay information, represented as a 3-dimensional tensor, transformed into a contextualized format that can be used for downstream prediction, including BSI.

4 Data Representation

In SARD, concept embedding plays a crucial role due to the categorical nature of the information related to hospital visits. Each piece of information is encoded as a 'word' and then transformed into a word embedding using techniques like Word2Vec. These word embeddings are aggregated to form a comprehensive visit embedding, summarizing the data from a single hospital visit. However, here we are modeling EHR data which predominantly consists of numerical information derived from charts and monitoring devices. This data can be best characterized as a series of signals, reflecting real-time physiological changes in patients. Directly applying concept embedding, as done in the SARD model, would lead to a substantial loss of critical information.

To effectively represent this dynamic and complex nature of EHR data, RatchetEHR adopts a distinct approach, inspired by the method outlined by Wang et al. [10]. Here, we represent the data in a structured form. We segment each patient's ICU visit into discrete time intervals, termed as timeframes. Each timeframe encompasses the data relevant to its respective time interval. In constructing these time-frames, we categorize our data into two distinct types: numerical and categorical. Numerical data primarily includes readings from monitors and results from examinations – for example, respiratory rate and blood pressure. On the other hand, categorical data includes aspects such as diagnoses, which we represent using one-hot encoding vectors.

For each timeframe, we then use the numerical and categorical vectors to form a singular, comprehensive input embedding that represents that specific timeframe. This process is repeated for each interval, building a sequential representation of a patient's ICU stay. The result of this process is a two-dimensional matrix for each patient's ICU visit, constructed by concatenating these time-frames in chronological order. This methodology allows RatchetEHR to maintain the integrity of both the numerical and categorical data, capturing the dynamic and complex nature of EHR data for each patient's ICU stay.

Formally, the framework splits each ICU stay into time frames (notated as TF) of h hours. Each time frame j of ICU stay i can be viewed as the following vectors:

$$u_{ij} = (u^i_{j1}, u^i_{j2}, \ldots, u^i_{jk}) \qquad w_{ij} = (w^i_{j1}, w^i_{j2}, \ldots, w^i_{jm})$$

where v^i_{jk} is the median of all values of feature k at time frame j for ICU-stay i, and w^i_{jm} is an indicator for whether the code number m occurred for ICU-stay i at time frame j. Therefore, $(w^i_{j1}, w^i_{j2}, \ldots, w^i_{jm})$ is a BOW (bag of words).

We will denote $l = k + m$ and use || as the concatenation operation.

Therefore,

$$v_{ji} = u_{ij} || w_{ji} \in R_l$$

Each sample i can be viewed as a 2-dimensional Tensor $\mathbf{W}^i \in \mathrm{R}^{p \times l}$, where p is the number of time frames for the i sample. The rows of the tensor are the time frames of the ICU-stay i, which can be denoted more formally:

$$\mathbf{W}_{ji} = V_{ji}$$

These samples are the input to the framework model.

4.1 Model Architecture

Two primary distinctions set RatchetEHR apart from the SARD model. First, while SARD aggregates visit data into a singular sum of concept codes, obscuring potential hidden structures within a visit, RatchetEHR reincorporates the Graph Convolutional Transformer (GCT) component, as suggested by Choi et al. [7]. This integration allows for a more nuanced understanding of the data.

The architecture of RatchetEHR, detailed in Fig. 1, includes the following components:

Time Frame Embedding. As introduced in Sect. 3, it is a 3-dimensional tensor input representation for the ICU data. It is composed of timeframes of h hours of the EHR data, a signal-like format suitable for time-series analysis.

Temporal Embedding. Utilizing fixed positional encoding, maintains the chronological order of the timeframes within the ICU stay, crucial for preserving the sequential nature of the data. Contrary to Kodialam et al. [8], where the visits do not occur regularly, the time frames are regular. Therefore, we used the fixed positional encoding introduced in the original Transformer article [6]:

$$PE_{(pos,2i)} = sin(\frac{pos}{10000^{\frac{2i}{T}}}) \qquad PE_{(pos,2i+1)} = cos(\frac{pos}{10000^{\frac{2i}{T}}})$$

Where pos is the position and i is the dimension.

Transformer Encoder. To effectively handle the sequential and time-variant nature of EHR data, we utilize the Transformer model. It is adept at creating contextualized embeddings that correlate with other time-frames. This is critical as most features at a certain time are dependent on their previous values. We include K transformer encoder layers, as delineated in Vaswani et al. [6], to output contextualized time frame embeddings.

Learned Time-frame Embedding. To mitigate the risk of overfitting in tasks with limited EHR data, we incorporated a learned time-frame embedding for each input. This concept, inspired by the BERT [CLS] token [11], aids in reducing the parameter count for downstream task predictions by encapsulating essential information within a trainable parameter.

MLP. A feed-forward neural network that transforms the learned timeframe embedding into a probability that indicates the class of the sample.

4.2 Model Refinements

To refine the RatchetEHR architecture, we integrated several advanced methods, addressing key challenges encountered during the experimentation with our SARD-inspired base model. A primary concern was the tendency of Transformer models, due to their large parameter count, to overfit, especially when dealing with limited sample sizes typical of EHR prediction tasks. Despite this, their strength in processing the sequential nature of EHR data is undeniable.

Transfer Learning. We adopted a dual-stage approach, akin to the BERT model's methodology [11], consisting of pre-training and fine-tuning stages. The pre-training phase employs a masking task on the extensive EHR data samples to initialize the model's weights, which is crucial for enhancing performance in downstream tasks. The architecture is augmented with a linear layer, projecting the contextualized time frame embeddings back into the input initial representation.

GCT Component. We integrated a Graph Convolutional Transformer (GCT) component into RatchetEHR to address another baseline model limitation: the lack of consideration for inter-feature relationships. By modeling these relationships as a hidden graphical structure, the GCT component enhances the robustness of time-frame embeddings in [7]. Trained across numerous time-frame embeddings, this method not only boosts the model's ability to discern hidden structures but also helps in circumventing overfitting.

Sampler and Focal Loss. To address class imbalance and its resultant prediction bias and overfitting, we implemented a weighted sampler for oversampling, creating more balanced mini-batches for training. The class weight, w_i, is inversely proportional to the class frequency, n_i. Additionally, focal loss is employed, focusing the model's learning on more challenging examples, a technique widely used in object recognition tasks with high class imbalance.

Child Tuning. To further enhance the model's efficiency and reduce overfitting, Child Tuning, as described in [12], is employed. This method limits the training to only the most relevant parameters identified through Fisher information, which assesses the sensitivity of the model to changes in each parameter. This selective training approach streamlines the model, focusing on parameters most critical to the task at hand.

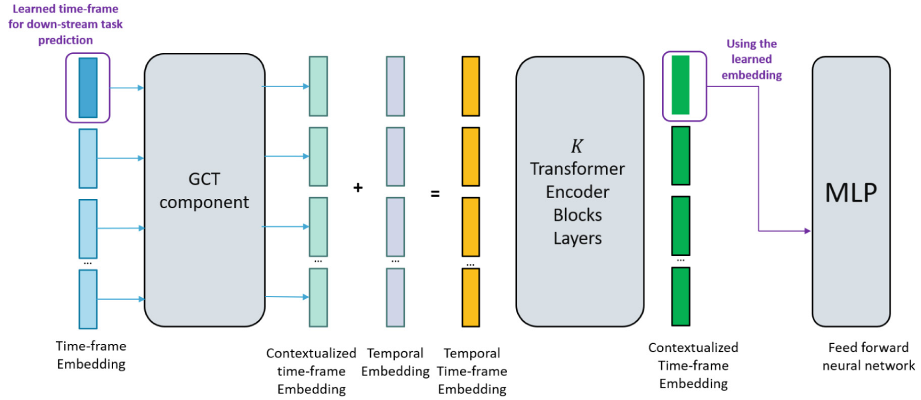

Learned time-frame for down-stream task prediction
GCT component
+
=
K Transformer Encoder Blocks Layers
Using the learned embedding
MLP
Time-frame Embedding
Contextualized time-frame Embedding
Temporal Embedding
Temporal Time-frame Embedding
Contextualized Time-frame Embedding
Feed forward neural network

Fig. 1. RatchetEHR architecture. The architecture has three key components: Time Frame Embedding, Temporal Embedding, and Transformer Encoder. The integration of the Graph Convolutional Transformer (GCT) component is also depicted, highlighting its role in enhancing the ability of the model to identify hidden structural relationships within the data. Advanced methodologies such as Transfer Learning, Learned Time-frame Embedding, Focal Loss, and Child Tuning are incorporated to optimize the performance of the model, particularly in addressing challenges like limited sample sizes, class imbalance, and overfitting.

5 Experiments

5.1 Datasets

We utilized the publicly available dataset MIMIC-IV which contains deidentified EHR data from 50,934 patients and 73,141 ICU stays at Beth Israel Deaconess Medical Center between 2008–2019. MIMIC-IV provides a rich source of EHR information, including vital signs, medication records, laboratory results, and patient demographics, acquired by routine clinical care, monitors and more. We utilized the ICU module of the MIMIC-IV dataset. This module provides detailed information about individual patient visits to the ICU, including subject ID, start and end times of the ICU stay, and various medical measurements and events recorded during the stay. The data was stored in a PostgreSQL database.

To facilitate data extraction and manipulation, RatchetEHR employs SQLAlchemy, a Python-based SQL toolkit, to generate patient cohorts and retrieve relevant feature information from the PostgreSQL database. This extraction process focuses on critical ICU metrics such as vital signs, medication records, laboratory results, and patient demographics, ensuring a comprehensive dataset for analysis.

Data preparation involved cleaning processes to handle missing values and outliers in the dataset. We employed linear interpolation provided by pandas framework to address gaps in the data and utilized established medical thresholds to identify and rectify out-of-range values [13].

To reduce the space of ICD-10 codes we utilized a mapping of ICD-10 codes to diseases as provided by [14], which helped streamline the dataset, making it more manageable and conducive for our analysis.

5.2 Prediction Task

Bloodstream Infection (BSI) is a critical condition that significantly impacts ICU patients, resulting in prolonged hospital stays, life-threatening complications, and notably high morbidity and mortality rates exceeding 30%, [1]. The standard diagnostic procedure for BSI involves a blood culture test, which typically requires one to two days to yield results. This delay is critical, considering the rapid progression of BSI and its severe consequences. Prompt detection and immediate antibiotic treatment are crucial for reducing the associated high morbidity and mortality rates, yet early-stage detection remains a challenge for physicians.

Our objective was to train RatchetEHR and other machine-learning models to predict BSI in patients who underwent a blood culture test and remained in the hospital for a minimum of two days post-test. This approach aligns with several existing studies aimed at forecasting BSI risk. Our architecture was specifically tested for its effectiveness in predicting BSI.

For cohort building, we replicated the method used by Roimi et al. [2], adhering to the guidelines set by the Center for Disease Control and Prevention (CDC)/National Health Safety Network (NHSN). We identified patients with BSI by detecting common commensal bacteria related to BSI, as listed in the NHSN organism tab. Patients showing growth of these bacteria in blood culture tests were labeled as positive for BSI. The detection was based on blood collection entries in the measurement table where the measurement attribute was blood culture.

Our selection criteria for ICU stays focused on cases where blood collection occurred more than 48 hours after admission. This criterion aimed to exclude patients who were admitted to the ICU primarily for surgical reasons, as they are generally not at risk for BSI. The study design, illustrated in Fig. 2, presents the timeline for each patient's hospital stay relative to the blood culture collection.

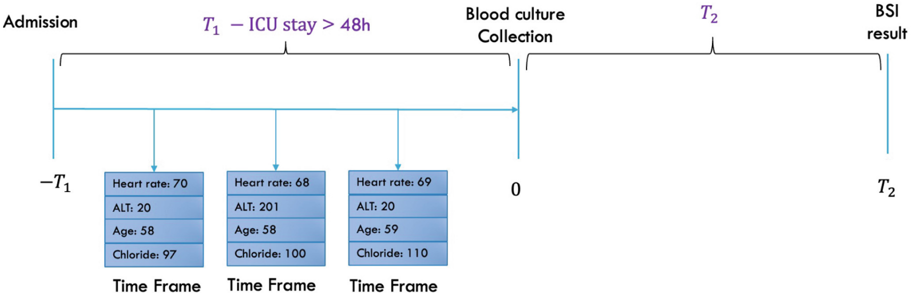


Fig. 2. Prediction task. The study design for each patient in the hospital. 0 is the index date which is the time of the blood culture collection. T_1 is the number of hours of the admission to the hospital before the blood culture test. During this interval, we collected the features. In the interval [0, T_2], we refrained from collecting data to prevent data leakage, as this is the period between the test and the results (the interval time is higher than 24 hours).

The inclusion criteria included undergoing blood culture test, duration of at least 48 hours at the ICU before blood culture test, and admission to the MICU (Medical ICU), SICU (Surgical ICU) and TSICU (Trauma ICU). In total, our cohort comprised 12,139 ICU stays, of which 538 were identified with BSI, representing a prevalence of 4.4%, highlighting the highly imbalanced nature of the dataset. It is important to note that BSI is sometimes treated without conclusive laboratory results and may not always be consistently coded in EHR data. This could potentially lead to an under-prediction of true BSI cases, which is a major limitation of using EHR data for this type of analysis.

Each sample consisted of at most 30 timeframes, with each timeframe encompassing four hours of data. This setup effectively captures a comprehensive timeline of five days leading up to each blood culture test. To ensure robustness in our model evaluation and to prevent data leakage associated with the year of ICU stay, we strategically split the dataset into training-validation and test sets based on the year of the ICU admission. Specifically, ICU stays from the years 2008 to 2017 were allocated to the training-validation set, while those from 2017 to 2019 were designated for the testing set. This split was chosen to ensure a sufficient sample size for training while maintaining a temporal separation between the training and testing data, which can help assess the model's performance on future, unseen data.

Given the highly imbalanced nature of the task, with a significantly lower prevalence of BSI cases, we opted to evaluate our model using the AUC-ROC score, a metric less sensitive to class imbalance. Additionally, considering the relatively small size of the sample dataset, we anticipated a high variance in performance metrics due to the initial split the data into training, validation, and test sets. To account for this variability and to ensure a comprehensive evaluation, we conducted 10 different experimental runs for each model. In each iteration, the training, validation, and test sets were randomly selected, providing a thorough and varied assessment of the model's performance across multiple splits of the dataset.

5.3 Benchmark Models

RNN. We employed an RNN model inspired by the architecture and hyperparameters detailed in [15]. The model uses Tanh activation and consists of two RNN layers with batch normalization. The final hidden state of the RNN feeds into a linear layer, projecting it to the probability of BSI occurrence in patients.

LSTM. Adapting the RNN model architecture, the LSTM variant replaces RNN layers with LSTM layers while maintaining the same hyperparameters. LSTM models are particularly adept at processing longer sequences, offering an improvement over traditional RNNs. However, it is important to note that LSTM models process inputs sequentially.

LSTM / RNN + Focal Loss. We also evaluated LSTM and RNN models trained with focal loss.

Focal loss is instrumental in directing the model's attention towards more challenging, incorrectly classified examples, thereby enhancing model performance on complex cases.

XGBoost. For the XGBoost classifier, as provided by the dmlc XGBoost package, we adapted our input data to fit the classifier's requirements. Given that XGBoost processes two-dimensional data, and our input is a three-dimensional matrix, we reduced the dimensionality by computing the median of the last eight timeframe embeddings (covering four hours) before the blood culture collection. Hyperparameter tuning was conducted using a randomized cross-validation search over 10,000 iterations, utilizing the scikit-learn framework.

Random Forest. In the case of the RandomForest classifier, hyperparameter optimization was achieved through a randomized cross-validation search with 10,000 iterations. We employed balanced class weights to address the class imbalance inherent in BSI prediction data.

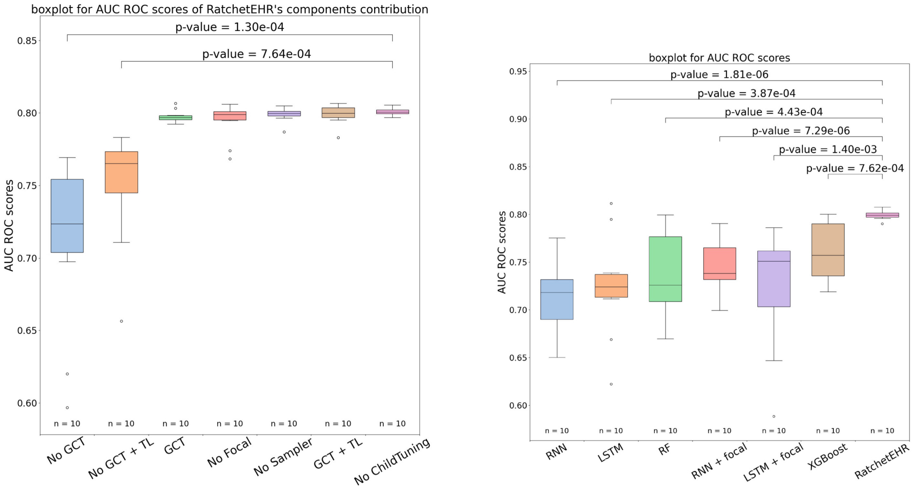
RatchetEHR Variations. Several variations of the RatchetEHR model were tested by modifying or removing specific components. For the transfer learning (TL) approach in the pretraining stage, the AdamW optimization algorithm was utilized, along with parameters such as a batch size of 32, dropout rate of 0.1, learning rate of 10^{-4}, and weight decay of 0.2. The fine-tuning stage hyperparameters were based on the BERT article [11], including the warmup method for learning rate adjustment. The learning rate gradually increases during the initial m steps of the optimization algorithm and then decreases linearly. A smaller batch size of 17 was used, along with a dropout of 0.5, an initial learning rate of $\alpha \cdot 10^{-3}$, and a weight decay of $\beta \cdot 0.3$.

5.4 Prediction Performance

We first assessed the contribution of the different components used in the RatchetEHR architecture. The analysis revealed that the incorporation of the GCT component substantially enhanced the performance of the prediction (Fig. 3. [A]). This improvement can be attributed to the ability of the GCT component to train on a vast number of timeframe embeddings, each representing a single timeframe. This extensive training leads to more robust and contextually enriched input embeddings that effectively capture the hidden graphical structure of EHR data features. Consequently, this contributes to mitigating the challenges posed by the class imbalance and the limited size of the dataset.

In the model variations lacking the GCT component, the transfer learning approach boosts performance and reduces the variance in the test set. This suggests that transfer learning is particularly effective in refining the model's accuracy in the absence of the GCT component. Despite small differences, the top-performing architecture was the one that included all components, including GCT, focal loss, sampler, and transfer learning, but without child tuning (average AUC-ROC = 0.8 ± 0.002). We suggest that this might be due to the substantial drop in the number of parameters that the child tuning masked and therefore were not updated.

Comparative analysis with other models highlights that while traditional models like RNN and LSTM show varying degrees of effectiveness, they are somewhat limited, especially in small datasets (Fig. 3. [B]). The application of focal loss function, commonly used in object detection tasks, shows improvements in these models, but not drastically.



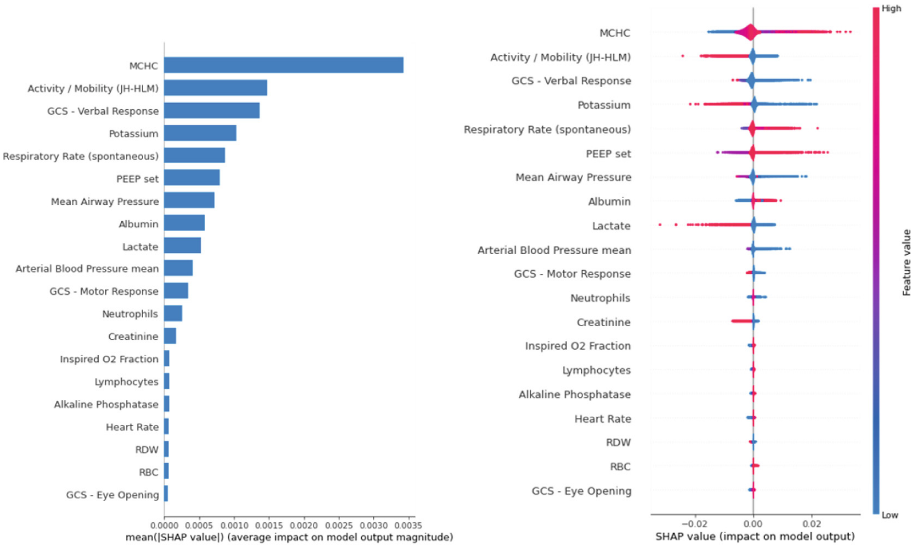
(a) (b)

Fig. 3. Evaluation of RatchetEHR performance. A. Boxplots show AUC-ROC on the test sets in the 10 iterations. We compared different variations of our architecture to showcase the relative contribution to the performance of each component. Values were compared using t-Test. The GCT component provided a significant boost to the performance. TL: transfer learning approach. **B.** Boxplots show AUC-ROC on the test sets in the 10 iterations. We compared different algorithms to the full version of RatchetEHR. Values were compared using t-Test. RF: Random Forest.

XGBoost, known for its proficiency with tabular data, shows higher performance, yet it is outperformed by RatchetEHR. This superior performance of RatchetEHR is largely due to its ability to process raw data through the transformer model, which efficiently discovers hidden structures and relationships within the time-frame embedding features and among the time-frames themselves.

6 Explainability

Machine learning models are prone to bias, confounders and other issues. This leads to mistrust among the users, especially in the healthcare domain, where the models' output affects the patients' quality of life. Many studies were conducted to provide insight to the model's decision making, to enhance the users' assurance on it. To provide a deeper understanding of how our RatchetEHR model arrives at its predictions, we utilized SHAP values. SHAP values offer an insightful way to interpret complex machine learning models, as explained in [16]. Here, we generated SHAP summary plots for 100 randomly selected ICU stays. Given the extensive number of features involved in each ICU stay, we focused on optimizing performance without compromising the depth of our analysis. We utilized the GradientExplainer, a component of the SHAP framework. This tool efficiently approximates SHAP values using expected gradients, as detailed in [17]. This approach enabled us to maintain computational efficiency while still providing rich, interpretable insights into the features driving the model's predictions (Fig. 4).



(a)

(b)

Fig. 4. Explainability of the model. A. SHAP summary bar plot, displaying the importance of each feature. **B.** SHAP summary violin plot, illustrating how feature values affect the model's predictions.

This analysis revealed that the most important feature in our prediction of BSI was Mean Corpuscular Hemoglobin Concentration (MCHC) (Fig. 3). Typically, MCHC is a measure of the average concentration of hemoglobin in a person's red blood cells, used primarily to diagnose and monitor conditions related to red blood cell health, such as anemia. This finding aligns with previous research where MCHC was identified as a relevant factor in BSI, as noted in studies by Roimi et al. [2] and Zoabi et al. [3], suggesting a potential, yet not fully explored, link between MCHC levels and BSI. Other features that our analysis underscored include the Glasgow Coma Scale (GCS) scores for verbal and motor responses, which echo the findings in Roimi et al. [2] and Mahmoud et al. [18]. Additionally, variables such as Albumin levels, respiratory rates, creatinine, and heart rate were also identified as significant, consistent with observations in the studies by Mahmoud et al. [18] and Zoabi et al. [3]. It is important to note that these correlations, while statistically significant in the context of the model, may not directly imply a causal relationship. Rather, it could reflect complex interplays in the patient's health status, where alterations in blood measurement levels coincide with factors that contribute to the susceptibility or onset of BSI.

7 Discussion and Conclusions

We presented here a complete framework for modeling EHR data of hospitalizations using a transformer-based architecture. We show that this framework provides superior performance over other state-of-the-art machine-learning approaches. This architecture is adept at effectively processing sequential EHR data, a crucial aspect given the temporal nature of medical records and their importance in clinical decision-making. This capability is particularly vital in predicting conditions like BSI, where the timing and evolution of patient data points are key indicators of the patient's health trajectory.

A pivotal aspect of RatchetEHR is the integration of the GCT component. By leveraging this component, RatchetEHR can uncover the hidden structural relationships within the data, crucial for understanding complex clinical scenarios. The GCT component notably enhances the ability to process and interpret each timeframe of EHR data. This capability is instrumental in the superior performance of our framework. Other components added to the architecture, including the focal loss, Sampler, and ChildTuning did not result in significant performance improvements, however, they were incorporated to address specific challenges such as class imbalance and overfitting. Future work could explore simplifying the model architecture to strike a balance between complexity and performance.

While complex, we show here that is it possible to extract feature importance from the model and provide the much-needed explainability. This aspect is highly valuable in clinical settings, where understanding the 'why' behind a model's prediction is as crucial as the prediction itself. This transparency allows clinicians to trust and effectively utilize AI-driven insights in their decision-making process.

It is important to consider potential limitations and biases inherent in EHR data. Inconsistencies in data collection, documentation, and coding practices across different healthcare systems may impact the model generalizability. For example, we are probably under-predicting cases of BSI due to inconsistent administrative coding and treatment without conclusive laboratory results. Future work could involve validating the model's performance on independent EHR datasets to assess its robustness and transferability, in addition to a prospective study in real-world scenarios.

In conclusion, this study contributes to the field of medical informatics by introducing an innovative approach to EHR data analysis and opens up new possibilities for future research to further enhance AI capabilities in healthcare.

Acknowledgments. DA is supported by the Azrieli Faculty Fellowship. We thank Joseph Bingham for the constructive review of the manuscript.

Data and Code Availability. MIMIC-IV was downloaded from the PhysioNet project: (https://physionet.org/content/mimiciv/2.2/. The code developed in this study is available at https://github.com/OrtalHirszowicz/RatchetEHR.

Funding Statement. This research received no specific grant from any funding agency in the public, commercial or not-for-profit sectors.

Competing Interests Statement. DA reports consulting fees from Carelon Digital Platforms. OH has no competing interest.

References

1. Bates, D.W., Pruess, K.E., Lee, T.H.: How bad are bacteremia and sepsis?: Outcomes in a cohort with suspected bacteremia. Arch. Intern. Med.. Intern. Med. **155**(6), 593–598 (1995)
2. Roimi, M., Neuberger, A., Shrot, A., Paul, M., Geffen, Y., Bar-Lavie, Y.: Early diagnosis of bloodstream infections in the intensive care unit using machine-learning algorithms. Intensive Care Med. **46**, 454–462 (2020)
3. Zoabi, Y., Kehat, O., Lahav, D., Weiss-Meilik, A., Adler, A., Shomron, N.: Predicting bloodstream infection outcome using machine learning. Sci. Rep. **11**(1), 20101 (2021)
4. Zhang, F., Wang, H., Liu, L., Teng, S., Ji, B.: Machine learning model for the prediction of gram-positive and gram-negative bacterial bloodstream infection based on routine laboratory parameters. BMC Infect. Dis. **23**(1), 675 (2023)
5. Choi, D.H., Lim, M.H., Kim, K.H., Shin, S.D., Hong, K.J., Kim, S.: Development of an artificial intelligence bacteremia prediction model and evaluation of its impact on physician predictions focusing on uncertainty. Sci. Rep.. Rep. **13**(1), 13518 (2023)
6. Vaswani, A., et al.: Attention is all you need. Adv. Neural Inf. Proc. Syst. **30** (2017)
7. Choi, E., et al.: Learning the graphical structure of electronic health records with graph convolutional transformer. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 606–613 (2020)
8. Kodialam, R., Boiarsky, R., Lim, J., Sai, A., Dixit, N., Sontag, D.: Deep contextual clinical prediction with reverse distillation. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, pp. 249–258 (2021)

9. Zisser, M., Aran, D.: Transformer-based time-to-event prediction for chronic kidney disease deterioration. J. Am. Med. Inform. Assoc.. Am. Med. Inform. Assoc. **31**, 980–990 (2024)
10. Wang, Y., Guan, Z., Hou, W., Wang, F.: Trace: Early detection of chronic kidney disease onset with transformer-enhanced feature embedding. In: VLDB Workshop on Data Management and Analytics for Medicine and Healthcare, pp. 166–182. Springer (2021). https://doi.org/10.1007/978-3-030-93663-1_13
11. Devlin, J., Chang, M.-W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
12. Xu, R., et al.: Raise a child in large language model: towards effective and generalizable fine-tuning. In: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, Online and Punta Cana, Dominican Republic, pp. 9514–9528. Association for Computational Linguistics, November 2021
13. Wang, S., McDermott, M.B.A., Chauhan, G., Ghassemi, M., Hughes, M.C., Naumann, T.: MIMIC-extract. In: Proceedings of the ACM Conference on Health, Inference, and Learning. ACM, April 2020
14. Kuan, V., et al.: A chronological map of 308 physical and mental health conditions from 4 million individuals in the english national health service. Lancet Digit. Health **1**(2), e63–e77 (2019)
15. Boner, Z., Christopher, M., Nguyen, N.: Deep learning risk prediction of bloodstream infection in the intensive care unit. arXiv preprint arXiv:2209.14546 (2022)
16. Lundberg, S.M., Lee, S.-I.: A unified approach to interpreting model predictions. In: Guyon, I., et al. (eds.) Advances in Neural Information Processing Systems, vol. 30, pp. 4765–4774. Curran Associates, Inc. (2017)
17. Sundararajan, M., Taly, A., Yan, Q.: Axiomatic attribution for deep networks. In: International Conference on Machine Learning, pp. 3319–3328. PMLR (2017)
18. Mahmoud, E., Al Dhoayan, M., Bosaeed, M., Al Johani, S., Arabi, Y.M.: Developing machine-learning prediction algorithm for bacteremia in admitted patients. Infect. Drug Resist., 757–765 (2021)

Modeling Multiple Adverse Pregnancy Outcomes: Learning from Diverse Data Sources

Saurabh Mathur[1](✉), Veerendra P. Gadekar[2], Rashika Ramola[3], Peixin Wang[3], Ramachandran Thiruvengadam[4], David M. Haas[5], Shinjini Bhatnagar[6], Nitya Wadhwa[6], Garbhini Study Group[6], Predrag Radivojac[3], Himanshu Sinha[2], Kristian Kersting[7], and Sriraam Natarajan[1]

[1] The University of Texas at Dallas, Richardson, TX, USA
saurabhsanjay.mathur@utdallas.edu
[2] Indian Institute of Technology Madras, Chennai, Tamil Nadu, India
[3] Northeastern University, Boston, MA, USA
[4] Pondicherry Institute of Medical Sciences, Puducherry, India
[5] Indiana University School of Medicine, Indianapolis, IN, USA
[6] Translational Health Sciences and Technology Institute, Faridabad, India
[7] Technische Universität Darmstadt, Darmstadt, Germany

Abstract. We consider the problem of modeling adverse pregnancy outcomes (APOs) from diverse data sets and aim to understand what is common between them and what is unique for each of these data sets. To this effect, we consider three different data sets (a clinical study from the US, EHRs from a US hospital, and a clinical study in India) and model three specific APOs - preterm birth, new hypertension, and preeclampsia. Since LLMs can efficiently summarize the scientific literature, we use them to generate initial hypotheses and use the different data sets to refine the hypotheses to create joint probabilistic models (as Bayesian networks). Our analyses show that there are eight relationships between risk factors common to all three populations and some unique relationships for specific populations.

Keywords: Bayesian Networks · Theory Refinement · LLMs

1 Introduction

Adverse Pregnancy Outcomes (APOs) such as preterm birth (PTB) pose a significant challenge in maternal-child health, with approximately one in ten births occurring prematurely on a global scale. The implications of PTB extend beyond immediate neonatal mortality, influencing both short-term and long-term health outcomes [17]. However, the relationship between APOs and their risk factors can vary across geographical regions [9]. This makes integration and analysis of multiple data sets vital to understanding APOs and mitigating their risk.

We aim to model the differences and commonalities between data sets of APOs from different countries. Specifically, we aim to perform this analysis by inducing interpretable probabilistic models from three data sets from 2 countries,

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 293–302, 2024.
https://doi.org/10.1007/978-3-031-66538-7_29

namely India (Garbh-Ini [1]) and the United States (nuMoM2b [13] and EHR data from Regenstrief Institute). This would help advance our understanding of the multifaceted nature of APOs and potentially inform targeted interventions tailored to specific geographical regions.

Probabilistic graphical models such as Bayesian networks [14,19] have long been used in AI for modeling interactions of multiple factors by learning joint distributions. In contrast to discriminative learning methods where the goal is to best predict an outcome, these generative models learn a joint distribution that can allow us to query comprehensively and understand the data in a more holistic manner. The biggest barrier to learning these models is the amount of data required which can be offset by using domain knowledge to construct an initial model and refining this model using the data.

Consequently, we employ the use of LLMs to generate an initial model (since LLMs can efficiently summarize the literature), refine the model with domain experts, and then use each of the data separately to refine the models for the respective populations. Once these different models are obtained, we perform meta-analyses of these models and summarize the findings. The common influence relationships that exist in all the data sets are between the risk factors BMI and HiBP and the three APOs new hypertension (NewHTN), preeclampsia (PreEc), and preterm birth (PTB). We also present the edges that are unique to each of these subpopulations (for instance, age is important in nuMoM2b but is not as influential in Garbh-Ini). Our hypothesis is that given such a unified yet diverse view, it is now possible to develop population-specific treatment plans for mitigating the APOs.

1.1 Data Description

nuMoM2b: The nuMoM2b (Nulliparous Pregnancy Outcomes Study: Monitoring Mothers-to-Be [13]) study focuses on identifying risk factors for APOs in the United States. It enrolled a diverse cohort of 10,038 nulliparous subjects across 8 US sites. Data collection occurred at the start of pregnancy and at subsequent visits throughout the pregnancy.

Electronic Health Records: Apart from the data from the nuMoM2b study, we also acquired Electronic Health Records (EHR) from the Regenstrief Institute. This data set includes non-nulliparous subjects but does not include information about family history of chronic conditions.

Garbh-Ini: The Garbh-Ini study [1] conducted in a single site within Haryana, India, aims to characterize PTB and identify associated risk factors. It enrolled 8,050 subjects both nulliparous and non-nulliparous, and collected data at the start of pregnancy and at subsequent visits throughout the pregnancy.

2 Background

Bayesian Networks (BNs [19]) are a class of Probabilistic Graphical Models (PGMs [14]) that factorize the joint distribution over a set of variables using a

Directed Acyclic Graph (DAG) and local conditional probability distributions (CPDs). The DAG has a node corresponding to each variable and a directed edge between nodes represents influence. For example, an edge Age → PTB would imply that the age of the subject at pregnancy influences our belief about the likelihood of preterm birth. The local CPDs quantify the influence in terms of probability values. Formally, a BN $\mathcal{M}$ over a set of n variables $V = \{X_1, \ldots, X_n\}$ is defined as the tuple $\langle \mathcal{G}, \theta \rangle$ where $\mathcal{G}$ is the DAG representing the structure of the BN and θ is the set of parameters for the local CPDs. The joint probability distribution over V defined by the BN is

$$P(X_1, \ldots, X_n) = \prod_{X \in V} P_\theta(X \mid \mathrm{Pa}_X) \tag{1}$$

where Pa_X is the set of parents of the BN node corresponding to variable X. BNs can reason under uncertainty and answer probabilistic queries about the variables. Additionally, since BNs consist of directed influences between variables and local conditional probabilities, they are easy to interpret.

The structure of the BN encodes conditional independence relations (CIs) between variables; each variable X is independent of its nondescendents given its parents Pa_X. These two properties – reasoning under uncertainty and interpretability make BNs a good fit for high-stakes domains such as healthcare that require models that can reason about complex relationships between variables while being able to develop trust with domain experts.

In this work, we induce BNs from each of the 3 data sets and compare the influence relations between APOs and their risk factors. However, inducing the structure of a BN directly from data is a data-hungry and computationally hard problem [7]. One approach to mitigate this problem is Theory Refinement [16]. This approach involves constructing an initial BN structure from domain knowledge and then refining this BN using data. Specifically, the BN is refined by performing local operations such as adding an edge, deleting an edge, and reversing an edge to maximize a given heuristic score. Commonly used scores include the Minimal Description Length (MDL [15]) and Bayesian-Dirichlet Scores(BD [6]). The MDL score can be adapted to exploit local structure [12] in the form of context-specific independence relations (CSIs [3]) if the local conditional distributions of the BN are represented as decision trees. While prior works obtain the initial BN from a domain expert, we aim to construct the initial BN by extracting approximate domain knowledge from a deep generative model.

Large Language Models as Approximate Knowledge Sources: LLMs [26] are a class of deep generative models for text data. They consist of two Artificial Neural Networks (ANNs) called an encoder and a decoder. These encoder and decoder ANNs are used to encode input prompt text from a user and to generate response text from the encoded prompt respectively. Examples of such LLMs include General Purpose Transformer (GPT [5]) and Gemini [23]. These models are fit using large amounts of textual data and have been shown to generate realistic text. However, they cannot reason about the information embedded in them [25]. As a result, prior work has tried to extract knowledge from existing

LLMs and inject the knowledge into models that can perform reasoning [10,18,20]. Inspired by these directions, we extract knowledge in the form of influence relations from an LLM, use this knowledge to instantiate a BN, and then refine the BN using clinical data.

3 Methodology

We aim to find the relationships between variables common across the three data sets, and the ones unique to each data set. We formalize this task as the following problem

Given: Data sets $\mathcal{D}_1, \mathcal{D}_2, \mathcal{D}_3$ over a set of variables $\boldsymbol{X}$, and an LLM $\mathcal{O}$
To Do: Find a set of pairwise influences that are supported in all 3 data sets and the sets of influences supported only in particular data sets.

We address the problem of identifying consistent and dataset-specific relationships by learning three BN structures and then comparing them. We identify two types of edges, common edges, present in all refined BNs, which represent pairwise influences consistently supported by all data sets; and dataset-specific edges, unique to a specific BN, which represent pairwise influences supported only in the corresponding data set.

However, learning a BN structure from data is a difficult problem. Firstly, BNs are highly expressive models, and finding the structure that maximizes the data likelihood would result in an overly complex BN that overfits that training data. To address this, BN structures are learned by minimizing a cost function that includes implicit [6] or explicit [15] regularization. Secondly, even with a cost function (say $\text{Cost}(\mathcal{M}, \mathcal{D})$), learning the structure of a BN requires us to solve the following combinatorial optimization problem:

$$\arg\min_{\mathcal{M}} \text{Cost}(\mathcal{M}, \mathcal{D}) \tag{2}$$

This problem requires a search over a superexponential number of BN graph structures. Not only is searching over such a large space computationally intractable (NP-Hard to be specific), but it also requires a large amount of data to be able to determine the optimal structure [7].

One way to make this problem tractable is to exploit domain knowledge. We encode domain knowledge in three ways. Firstly, we encode domain knowledge through the choice of scoring function. Specifically, we use the MDL scoring function that prefers concise structures over more complex structures through an explicit penalty term. Secondly, we use domain knowledge about relations between the variables to construct an initial BN structure. While such BN structures are generally elicited from domain experts, we obtain the initial structure by querying an LLM. We further restrict the search space by using domain knowledge to identify and exclude temporally impossible edges. For instance, the edge PTB → BMI is invalid as preterm birth cannot influence Body Mass Index (BMI) measured at the pregnancy's start. By incorporating domain knowledge, we restrict the search from an exhaustive exploration of all BN structures to a local search over the structures in the neighborhood of an initial structure obtained from an LLM.

3.1 BN Refinement Using the MDL Score

We refine the initial BN structure for each data set by minimizing the MDL score. The MDL score for a BN (denoted by $\mathcal{M}$) with respect to a data set (denoted by $\mathcal{D}$) is the sum of the description length of the data encoded using the BN model ($\text{DL}(D \mid \mathcal{M})$) and the description length of the BN model itself ($\text{DL}(\mathcal{M})$). Concretely, the MDL score is given as

$$\text{MDL}(\mathcal{M}; \mathcal{D}) = \text{DL}(D \mid \mathcal{M}) + \text{DL}(\mathcal{M}) \tag{3}$$

The first term is the description length of the encoded data and captures the number of bits required to encode the data points using the probabilities estimated by the BN model. The second term is the description length of the model and captures the complexity of the BN itself. Since Huffman coding allows data points to be encoded using their probabilities, $\text{DL}(\mathcal{D} \mid \mathcal{M})$ is approximated by the negative log-likelihood of the data set under the BN. The description length (DL) of a BN, denoted by DL(M), captures the complexity of the model. It consists of two components, the description length of the graphical structure of the BN $\mathcal{G}$ and that of the parameters of the local conditional distributions θ.

3.2 Encoding the BN Model

Description Length of the Graphical Structure. This term represents the space required to encode the BN's structure G. Each node's description includes the number of parents and their names. Since each node can be encoded in $\log n$ units of space, the description length of the structure is $\sum_{X \in V}(1 + |\text{Pa}_X|) \log n$.

Description Length of the Parameters. This term represents the space required to encode the parameters, θ. These parameters define the local CPDs over each node given its parents. There are two ways to encode these distributions, as tables and as trees. Conditional Probability Tables (CPTs) explicitly enumerate the conditional probability values corresponding to each parent configuration. Each entry in a CPT can be encoded as an ordered list of fixed-width floating-point values, each of which can be encoded in space $\frac{1}{2} \log N$, where N is the size of the data set. The resulting description length for all the CPTs of the BN is $\sum_{X \in V}(|X| - 1)|\text{Pa}_X|(\frac{1}{2} \log N)$.

Local conditional distributions can be represented as trees to exploit local structure [12] in the form of CSIs [3]. The description length of such a tree-structured local conditional distribution over a variable X given its parents is $B(|X| - 1)(\frac{1}{2} \log N) + \sum_{l=1}^{d} \log(|\text{Pa}_X| - A_l)$. Here, B is the number of leaf nodes, d is the depth of the tree and A_l is the number of internal nodes at level l.

3.3 Computing the CSI-Aware MDL Score

To account for CSIs in the MDL score we use the Classification and Regression Trees (CART [4]) algorithm. At each node, we fit a decision tree to predict the node's value from its parents. This decision tree serves as the tree-structured CPD for computing the MDL score. The overall MDL score is given by the following equation:

$$\begin{aligned} \mathrm{MDL}(M;\mathcal{D}) = &-\sum_{\boldsymbol{x}\in\mathcal{D}} \log P_{\mathcal{M}}(\boldsymbol{x}) + \sum_{X\in V}(1+|\mathrm{Pa}_X|)\log n \\ &+ \sum_{X\in V} B_X(|X|-1)(\frac{1}{2}\log N) + \sum_{l=1}^{d}\log(|\mathrm{Pa}_X| - A_{X_l}) \end{aligned} \tag{4}$$

where B_X and A_{X_l} are the number of leaf nodes and the number of internal nodes at level l for the decision tree fit for node X respectively.

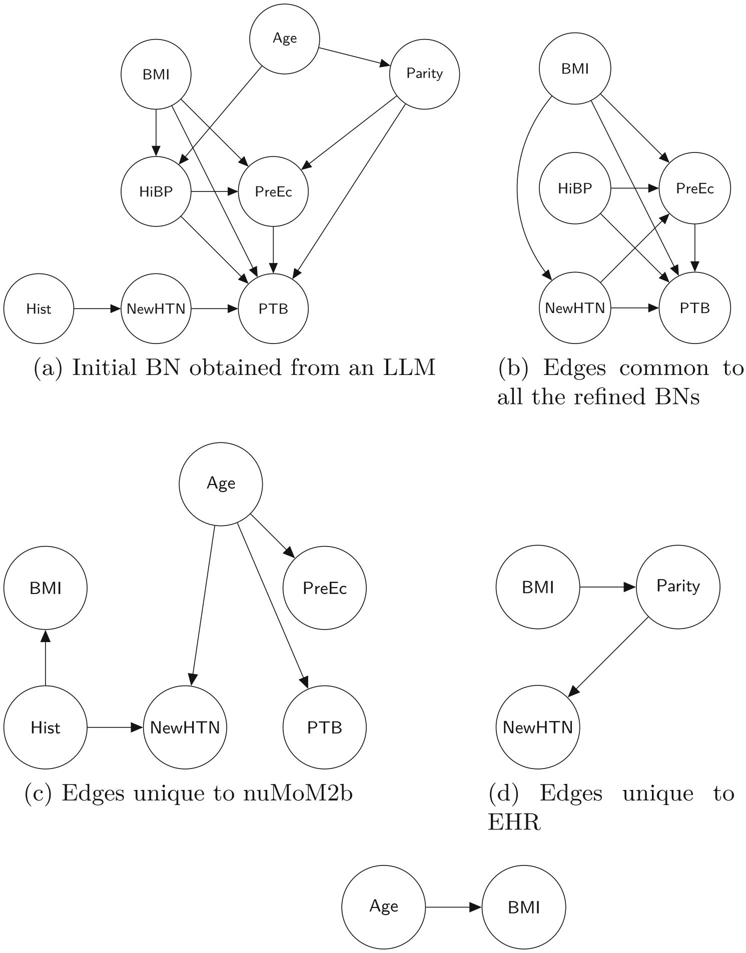
4 Experimental Evaluation

We consider 3 APOs, namely, New Hypertension (NewHTN), Preeclampsia (PreEc), and Pre-term birth (PTB), and study their relationship with 5 risk factors from prior work [8]. Specifically, the risk factors include Family History of diabetes (Hist), Age at the start of pregnancy (Age), Body Mass Index at the start of pregnancy (BMI), presence of Hypertension at the start of pregnancy (HiBP), and Parity. Of these variables, Parity does not apply to nuMoM2b as the study selected nulliparous subjects (Parity = 0) and Hist was not available in the EHR data. We removed data points that had missing values for any of the considered variables. Table 1 summarizes the variables, their discrete values, and the corresponding proportions in each of the three data sets.

We obtained a set of edges from Gemini to construct an initial BN structure and then refined this structure for each of the three data sets. Figure 1 shows the initial BN obtained from Gemini, the edges common to all the refined BNs, and the edges unique to each of the three data sets[1]. Apart from these, the edges $\{\mathrm{Age} \rightarrow \mathrm{Parity}, \mathrm{Parity} \rightarrow \mathrm{PTB}, \mathrm{Parity} \rightarrow \mathrm{PreEc}\}$ were present in both the data sets that had the Parity variables available (Garbh-Ini and EHR).

The edges common to all three refined BNs reflect existing domain knowledge. High BMI is known to increase the risk of Hypertensive disorders of pregnancy such as preeclampsia and new hypertension [2,21]. Hypertensive disorders of pregnancy are known to increase the risk of preterm birth [24]. Finally, hypertension at the start of pregnancy (HiBP) and new hypertension are known risk factors for preeclampsia [11].

[1] The code for the experiments, the LLM prompt, and the list of temporally impossible edges is available at https://github.com/saurabhmathur96/BN-Refinement.



(a) Initial BN obtained from an LLM

(b) Edges common to all the refined BNs

(c) Edges unique to nuMoM2b

(d) Edges unique to EHR

(e) Edges unique to Garbh-Ini

Fig. 1. The initial BN structure obtained from an LLM (a), edges common to the BNs refined on all 3 data sets (b), edges unique to nuMoM2b (c), EHR (d) and (e)

The edge from BMI to Parity in the BN learned from the EHR data might reflect the fact that high obesity negatively influences fertility [22]. This edge is supported by the EHR data which has the largest proportion of high BMI subjects. While BMI is expected to rise with an increase in Age, the influence relation is unique to the BN learned from the Garbh-Ini data set.

Table 1. Variable-value proportions for each of the three data sets

Variable	Value	nuMoM2b	Garbh-Ini	EHR
Age	≤ 21	21.03%	31.54%	9.87%
	21–35	72.36%	67.78%	75.27%
	>35	6.61%	0.67%	14.86%
BMI	≤ 18	3.39%	19.64%	1.12%
	18–25	51.29%	67.13%	31.53%
	>25	45.31%	13.22%	67.35%
Parity	=0	100%	48.50%	6.3%
	0–2	N/A	47.12%	67.36%
	>2	N/A	4.38%	26.34%
Hist	TRUE	20.55%	8.10%	N/A
HiBP	TRUE	2.84%	2.10%	9.37%
PReEc	TRUE	5.85%	3.80%	7.54%
NewHTN	TRUE	16.12%	3.40%	11.09%
PTB	TRUE	8.11%	12.80%	9.41%
Total		9,368	4,159	16,487

5 Discussion

A few important differences between the populations need to be pointed out. First, while the nuMoM2b study studied nulliparous subjects (first-time mothers), there were no such restrictions in the other two datasets. Second, the common risk factors and APOs were chosen across the different data for the purposes of this study. Consequently, APOs such as gestational diabetes were not considered as they were computed differently in the Garbh-Ini study. Thus, some of the relationships such as the influence of family history might include some hidden confounders (such as gestational diabetes). Exploring these issues remains an open problem. Finally, a variable such as race, a social construct, which plays an important role in a diverse dataset such as the EHR is not considered due to its absence in the single-state study in India.

Nonetheless, several common themes emerged. The influence of HiBP and BMI is quite significant across populations and data sets. It is clear that in nuMoM2b participants, age has a direct influence on PTB while in Garbh-Ini participants, age directly influences BMI (potentially through multiple pregnancies). It is important to understand the key differences in the data itself and these models provide a way of doing that. Future research could explore several avenues, including incorporating more data sets, identifying hidden confounders, understanding the similarities and differences in population, and extending these analyses to more global data sets. Finally, integrating multi-omic data, such as gene expression and proteomics data, alongside clinical data from diverse sources could offer deeper insights into the molecular pathways underlying APOs.

Acknowledgements. SM, RR, AW, DH, PR and SN acknowledge the support by the NIH grant R01HD101246. VG, RT and HS thank the Centre for Integrative Biology and Systems Medicine (IBSE) and Robert Bosch Centre for Data Science and Artificial Intelligence (RBCDSAI), IIT Madras for their support. KK acknowledges support by the European Union grant agreement no. 101120763 - TANGO.

References

1. Bhatnagar, S., Majumder, P.P., Salunke, D.M.: A pregnancy cohort to study multidimensional correlates of preterm birth in India: study design, implementation, and baseline characteristics of the participants. Am. J. Epidemiol. **188**(4), 621–631 (2019)
2. Bohiltea, R.E., et al.: Impact of obesity on the prognosis of hypertensive disorders in pregnancy. Exp. Ther. Med. **20**(3), 2423–2428 (2020)
3. Boutilier, C., Friedman, N., Goldszmidt, M., Koller, D.: Context-specific independence in bayesian networks. In: Proceedings of the Twelfth International Conference on Uncertainty in Artificial Intelligence, pp. 115–123 (1996)
4. Breiman, L., Friedman, J.H., Olshen, R.A., Stone, C.J.: Classification and Regression Trees. Wadsworth (1984)
5. Brown, T., et al.: Language models are few-shot learners. Adv. Neural. Inf. Process. Syst. **33**, 1877–1901 (2020)
6. Buntine, W.: Theory refinement on bayesian networks. In: UAI, pp. 52–60. Elsevier (1991)
7. Chickering, M., Heckerman, D., Meek, C.: Large-sample learning of bayesian networks is NP-hard. J. Mach. Learn. Res. **5**, 1287–1330 (2004)
8. Chu, H., Ramola, R., Jain, S., Haas, D.M., Natarajan, S., Radivojac, P.: Using association rules to understand the risk of adverse pregnancy outcomes in a diverse population. In: PACIFIC SYMPOSIUM ON BIOCOMPUTING 2023: Kohala Coast, Hawaii, USA, 3–7 January 2023, pp. 209–220. World Scientific (2022)
9. Dhaded, S.M., et al.: The causes of preterm neonatal deaths in India and Pakistan (purpose): a prospective cohort study. Lancet Glob. Health **10**(11), e1575–e1581 (2022)
10. Dietterich, T.: What's Wrong with Large Language Models and What We Should Be Building Instead (2024)
11. Dimitriadis, E., et al.: Pre-eclampsia. Nat. Rev. Dis. Primers **9**(1), 1–22 (2023). https://doi.org/10.1038/s41572-023-00417-6
12. Friedman, N., Goldszmidt, M.: Learning bayesian networks with local structure. In: Proceedings of the Twelfth International Conference on Uncertainty in Artificial Intelligence, pp. 252–262 (1996)
13. Haas, D.M., et al.: A description of the methods of the nulliparous pregnancy outcomes study: monitoring mothers-to-be (nuMoM2b). Am. J. Obstet. Gynecol. **212**(4), 539-e1 (2015)
14. Koller, D., Friedman, N.: Probabilistic Graphical Models: Principles and Techniques. MIT Press, Cambridge (2009)
15. Lam, W., Bacchus, F.: Using causal information and local measures to learn bayesian networks. In: UAI, pp. 243–250. Elsevier (1993)
16. Mooney, R.J., Shavlik, J.W.: A recap of early work on theory and knowledge refinement. In: AAAI Spring Symposium: Combining Machine Learning with Knowledge Engineering (2021)

17. Ohuma, E.O., et al.: National, regional, and global estimates of preterm birth in 2020, with trends from 2010: a systematic analysis. The Lancet **402**(10409), 1261–1271 (2023)
18. Pan, S., Luo, L., Wang, Y., Chen, C., Wang, J., Wu, X.: Unifying large language models and knowledge graphs: a roadmap. IEEE Trans. Knowl. Data Eng. **36**, 3580–3599 (2024)
19. Pearl, J.: Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann, San Francisco (1988)
20. Petroni, F., et al.: Language models as knowledge bases? In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pp. 2463–2473 (2019)
21. Roush, G.C.: Obesity-induced hypertension: heavy on the accelerator (2019)
22. Silvestris, E., De Pergola, G., Rosania, R., Loverro, G.: Obesity as disruptor of the female fertility. Reprod. Biol. Endocrinol. **16**, 1–13 (2018)
23. Team, G., et al.: Gemini: a family of highly capable multimodal models. arXiv preprint arXiv:2312.11805 (2023)
24. Wilson, D.A., Mateus, J., Ash, E., Turan, T.N., Hunt, K.J., Malek, A.M.: The association of hypertensive disorders of pregnancy with infant mortality, preterm delivery, and small for gestational age. In: Healthcare, vol. 12, p. 597. Multidisciplinary Digital Publishing Institute (2024)
25. Zhang, H., Li, L.H., Meng, T., Chang, K.W., Van den Broeck, G.: On the paradox of learning to reason from data. In: IJCAI (2023)
26. Zhao, W.X., et al.: A survey of large language models. arXiv preprint arXiv:2303.18223 (2023)

OptimalMEE: Optimizing Large Language Models for Medical Event Extraction Through Fine-Tuning and Post-hoc Verification

Yaoqian Sun, Dan Wu, Zikang Chen, Hailing Cai, and Jiye An(✉)

Zhejiang University, Hangzhou 310058, China
an_jiye@zju.edu.cn

Abstract. Medical event extraction (MEE) aims to identify and extract medical events mentioned in clinical notes, serving as a fundamental task for many clinical applications. Traditional solutions demand significant labor and sophisticated model design. The emerging large language models (LLMs) are considered to be potential on MEE. However, such methods have been shown to encounter issues related to accuracy, interpretability, and generalizability. In this paper, we propose OptimalMEE to optimize LLMs for MEE through fine-tuning and post-hoc verification. We leverage the LLM paired with a parameter-efficient fine-tuning mechanism and develop a four-step post-hoc verification process aimed at refining and interpreting the extracted events. Experimental results on multi-center datasets illustrate the strength of the proposed OptimalMEE on accuracy, interpretability, and generalizability, with 0.902 and 0.809 in MicroF1.

Keywords: Medical Event Extraction · Large Language Models · Information Extraction · Text Mining

1 Introduction

Clinical notes are the narrative part of electronic health records (EHRs), documenting the interactions between patients and doctors. With the rapid growth of EHRs, the traditional manual review method has proved to be time-consuming and burdensome. Consequently, the field of medical event extraction (MEE) receives significant attention [1, 2]. MEE is a pivotal task in clinical text mining, which is fundamental for many clinical applications, such as clinical decision support, and trial recruitment [3].

Current studies on automatic MEE are carried out either based on rules or deep learning algorithms. The former requires great effort to define the rules and is challenging to expand or generalize [4]. Deep learning algorithms could alleviate the burdens above with representation learning. However, they still cost adequate labeled data to fit the model and entail significant effort in model design with massive parameters [5]. More recently, with the prevalence of ChatGPT [6], generative large language models

Y. Sun and D. Wu—Two authors contributed equally to this work.

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 303–311, 2024.
https://doi.org/10.1007/978-3-031-66538-7_30

(LLMs) have been widely utilized in various NLP tasks including text mining. However, current studies suggest they encounter challenges in accuracy, interpretability, and generalizability [7, 8]. Further explorations of optimization are still lacking.

In this work, we endeavor to address the challenges by optimizing large language models for medical event extraction through fine-tuning and post-hoc verification (OptimalMEE). Experiment results on multi-center datasets indicate the improvement of the proposed OptimalMEE, proving it could be a possible solution for the efficient extraction of medical events in the resource-constrained clinical scenario using LLMs.

2 Materials and Methods

2.1 Data

A total of 505 clinical notes on medical history belonging to 158 patients with lung cancer were collected as MAIN-dataset, from a hospital in Shanxi, China. To intimate the low-source scenario of labeled data, 100 clinical notes are used for the training set, 15 for the development set, and 390 for the test set. The datasets are divided randomly. A GENERALIZATION-Dataset is also collected with 16 clinical notes on discharge summaries belonging to 16 patients with lung cancer from a hospital in Beijing, China. They are all used for the method generalization evaluation. All clinical notes were first deidentified before any further process. The study is conducted in compliance with pertinent guidelines and regulations (see Appendix A.1).

Under the clinicians' instructions, we select *Medication*, *Operation*, *Imaging*, and *Gene Test* as four types of medical events to extract and uniformly define the event elements of each type as *event_time*, *event_project*, and *event_conclusion.* The statistical details of the two datasets are shown in Table 1.

Table 1. Statistical details of two datasets.

Statistics	MAIN-Dataset	GENERALIZATION-Dataset
Total number of clinical notes	505	16
The average number of events per note	4.21	7.25
Maximum number of events in one note	9	27
Total number of @Imaging	759	63
Total number of @Operation	340	9
Total number of @Gene Test	77	10
Total number of @Medication	952	34

Two engineers with medical informatics background were recruited to label the data. To guarantee the inter-annotator reliability, another experienced engineer with clinical knowledge reviewed the annotated data to determine final results as the gold standard.

2.2 Methods

Figure 1 shows an overview of OptimalMEE. The pipeline receives a clinical note with prompt as input and produces the extracted medical event list as output.

To clarify the objective of the MEE task, we formulate the task instruction prompt and concatenate it with the input clinical note (see Appendix A.2). During the fine-tuning stage, we leverage QLoRA [9] to fine-tune the base LLM. It is a widely used PEFT method, which significantly reduces memory usage and enhances efficiency.

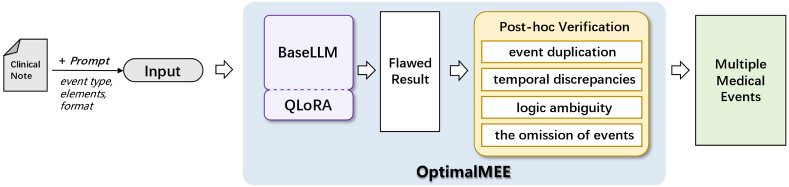


Fig. 1. Overview of OptimalMEE, where the QLoRA module is trained during fine-tuning.

While the fine-tuning aligned the model with the MEE task, the following issues remain in the raw events: *event duplication*, *temporal discrepancies*, *logic ambiguity*, and *the omission of events*. Thus, a four-step post-hoc verification process is devised to refine the results.

The deduplication self-checking is carried out by comparing elements of two events and merging them if the same. The issues of *temporal discrepancies* and *logic ambiguity* are addressed through cross-checking with the input. The *temporal discrepancies* could be detected and rectified by tracing back the *event_time* in the input. *Logic ambiguity* mainly arises from confusion regarding causes and conclusions. Since the conclusion should occur after the description of the event, it could be resolved by locating and comparing the positions of *event_project* and *event_conclusion* in the input.

The extracted *event_conclusion* may overlap with another event, which leads to *the omission of events*. To address this, we resort to recalling the fine-tuned LLM for detecting and re-extracting the missing events from the extracted *event_conclusion*. The detailed prompt is provided in Appendix A.2.

2.3 Experimental Setup

We instantiate OptimalMEE with ChatGLM3-6B [10] as the base model. It is worth noticing that our method also supports to adapt with other open-source LLMs.

We compare OptimalMEE against both traditional methods and LLM-based baselines. The formers include a manually defined rule-based extraction system and ReDEE [11], the state-of-the-art deep learning method for the document event extraction task. We also employ ChatGLM3-6B in three patterns: zero-shot learning (Only-LLM), few-shot learning (5-shot-LLM), and QLoRA fine-tuned (QLoRA-LLM).

Precision, recall, and F1 score at the event level and the micro average F1 score (MicroF1) across all types of events are selected as evaluation metrics.

3 Results and Discussion

Table 2 shows the experiment results. All experiments were conducted on the MAIN-Dataset, except for the generalization study was on the GENERALIZATION-Dataset.

Results of comparison experiments prove that the native LLM could lead to a considerable improvement over traditional methods, indicating the great potential of LLM-based methods. Also, the fine-tuned LLM surpasses ReDEE and Only-LLM, stressing the remarkable efficiency of the PEFT algorithm. It is also worth noting that the 5-shot-LLM performs worse than only-LLM, which might be caused by verbosity [8].

The ablation study is conducted by removing one procedure at a time. According to Table 2, the steps of event re-extraction and time verification contribute most to the improvement, indicating it is still challenging for LLMs to distinguish overlapping medical events and the hallucination phenomenon of LLMs remains significant, emphasizing the essentiality of post-processing. The results also show that the duplication issue mainly occurs in *Imaging* events. This might be associated with the extensive nature of imaging conclusions, leading the model to interpret it as two distinct events.

Generalization results indicate the universal applicability of our proposed method with LLM and non-expression-based post-hoc verification when handling clinical notes with diverse writing styles and varying distributions of medical events (see Table 1).

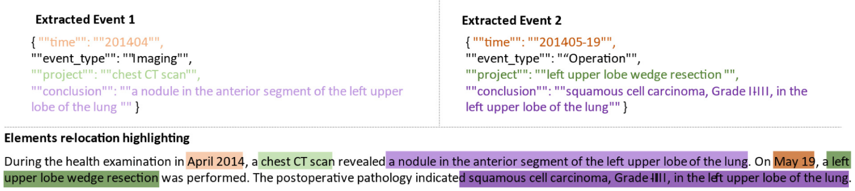
From the perspective of interpretability, Fig. 2 shows that post-hoc verification provides a double-check for the extracted result by tracing back to the input text and locating the extracted elements, which also serves as evidence for the extracted result.

Table 2. Experiment results, where 'w/o' stands for 'without'; 'Generalization' denotes OptimalMEE tests on the GENERALIZATION-Dataset.

Models	MicroF1	Operation			Imaging			Gene Test			Medication		
		Precision	Recall	F1 score	Precision	Recall	F1 score	Precision	Recall	F1 score	Precision	Recall	F1 score
Rule	0.513	0.625	0.714	0.667	0.447	0.514	0.478	0.727	1.000	0.842	0.717	0.223	0.340
ReDEE	0.415	0.250	0.312	0.278	0.727	0.121	0.208	0.188	0.328	0.239	0.400	0.563	0.468
Only-LLM	0.764	0.833	0.917	0.873	0.898	0.567	0.695	0.333	0.474	0.391	0.838	0.754	0.794
5-shot-LLM	0.592	0.944	0.756	0.840	0.739	0.472	0.576	1.000	0.438	0.609	0.902	0.354	0.508
QLoRA-LLM	0.804	0.803	0.770	0.786	0.740	0.767	0.753	0.360	0.450	0.400	0.892	0.851	0.871
OptimalMEE	**0.902**	**1.000**	**0.960**	**0.979**	**0.835**	**0.880**	**0.857**	**0.769**	**1.000**	**0.870**	**0.912**	**0.908**	**0.910**
w/o Duplication	0.890	1.000	0.960	0.979	0.817	0.880	0.847	0.704	0.950	0.809	0.916	0.874	0.895
w/o Omission	0.859	1.000	0.960	0.979	0.759	0.773	0.766	0.682	0.750	0.714	0.916	0.874	0.895
w/o Time	0.844	0.916	0.878	0.897	0.757	0.767	0.762	0.615	0.800	0.696	0.909	0.874	0.891
w/o Logic	0.888	0.930	0.892	0.910	0.818	0.862	0.840	0.769	1.000	0.870	0.919	0.905	0.912
Generalization	0.809	1.000	0.778	0.875	0.902	0.873	0.887	1.000	0.700	0.824	0.846	0.647	0.733

3.1 Limitations

In this study, though we evaluate our methods on multi-center datasets, the total sample size of testing is still limited. To provide more solid evidence of the improvement of



Extracted Event 1

```
{ ""time"": ""201404"",
""event_type"": ""Imaging"",
""project"": ""chest CT scan"",
""conclusion"": ""a nodule in the anterior segment of the left upper
lobe of the lung "" }
```

Extracted Event 2

```
{ ""time"": ""201405-19"",
""event_type"": "“Operation"",
""project"": ""left upper lobe wedge resection "",
""conclusion"": ""squamous cell carcinoma, Grade I-III, in the
left upper lobe of the lung"" }
```

Elements re-location highlighting

During the health examination in April 2014, a chest CT scan revealed a nodule in the anterior segment of the left upper lobe of the lung. On May 19, a left upper lobe wedge resection was performed. The postoperative pathology indicated squamous cell carcinoma, Grade I-III, in the left upper lobe of the lung.

Fig. 2. Example output and the corresponding locations in the input clinical note.

OptimalMEE, larger datasets should be considered. Besides, we only instantiate OptimalMEE with ChatGLM3-6B. Though the proposed method supports other open-source LLMs, the incremental gain and baseline LLM performance may differentiate across different base LLMs.

4 Conclusion

In this study, we proposed OptimalMEE which optimizes LLM for medical event extraction through fine-tuning and post-hoc verification. Concretely, we leverage QLoRA to fine-tune the base LLM with minimal annotation effort and develop a four-step post-hoc verification to improve the accuracy and interpretability of the extracted events and guarantee the generalizability of the method. Experiment results on multi-center datasets indicate the establishment of OptimalMEE provides a practical solution to MEE task and lays the groundwork for further applications in clinical information extraction.

Acknowledgments. This study was funded by the National Key Research and Development Program of China (grant number 2022YFF1203001).

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

A Appendix

A.1 The Relevant Legislation Statement

This study was a **secondary analysis of existing, anonymized data and not considered to be human subjects' research**. The need for ethics approval and informed consent is deemed unnecessary according to Article 32 of the “Approach to the Ethical Review of Life Science and Medical Research Involving Humans” issued by National Health Commission of Peoples' Republic China, Ministry of Education of Peoples' Republic China, Ministry of Science and Technology of Peoples' Republic China, and National Administration of Traditional Chinese Medicine of Peoples' Republic China in February 2023.

The specific content of the Article is translated as follows:

Approach to the Ethical Review of Life Science and Medical Research Involving Humans

Chapter 3 Article 32: Using human information data or biological samples to carry out life science and medical research involving humans in the following circumstances, which doesn't do harm to humans and involve sensitive personal information or commercial interests, can exempt ethical review to reduce the unnecessary burden on researchers and promote the development of life science and medical research involving humans.

(1) A research using overt data obtained legally, or **the data that is generated by observation and does not interfere with public behavior**;

(2) A research using **anonymous information data**;

Source document URL: https://www.gov.cn/zhengce/zhengceku/2023-02/28/content_5743658.htm.

A.2 Prompt Details

See Table A3.

Table A3. Prompt templates of different methods.

Method	Prompt Template
Zero-shot Learning	Please complete the task of extracting medical event information from the provided clinical note: The given clinical note includes clinical events belonging to the following four types: #Imaging, Gene Testing, Operation, Medication#. Please return the extracted event time, event type, event project name, and conclusion for the medical event belonging to #Imaging, Gene Testing, Operation, Medication# in JSON format: {"time": "","event_type": "","project": "","conclusion": ""}; if any information item is empty, please fill it with NONE. Do not create events or times on your own The clinical note to extract is as follows: """*{text}*"""

(*continued*)

Table A3. (*continued*)

Method	Prompt Template

Few-shot Learning (5-shot)	Please complete the task of extracting medical event information from the provided clinical note following examples provided: The given clinical note includes clinical events belonging to the following four types: #Imaging, Gene Testing, Operation, Medication#. Please return the extracted event time, event type, event project name, and conclusion for the medical event belonging to #Imaging, Gene Testing, Operation, Medication# in JSON format: {"time": "","event_type": "","project": "","conclusion": ""}; if any information item is empty, please fill it with NONE. Do not create events or times on your own
	Task examples are as follows:
	***Example 1: *{example}*
	Example 2: *{example}*
	Example 3: *{example}*
	Example 4: *{example}*
	Example 5: *{example}****
	The clinical note to extract is as follows:
	"""*{text}*"""

OptimalMEE (input)	Please complete the task of extracting medical event information from the provided clinical note: The given clinical note includes clinical events belonging to the following four types: #Imaging, Gene Testing, Operation, Medication#. Please return the extracted event time, event type, event project name, and conclusion for the medical event belonging to #Imaging, Gene Testing, Operation, Medication# in JSON format: {"time": "","event_type": "","project": "","conclusion": ""}; if any information item is empty, please fill it with NONE. Do not create events or times on your own The clinical note to extract is as follows: """*{text}*"""

(*continued*)

Table A3. (*continued*)

Method	Prompt Template
OptimalMEE (recalling during post-hoc verification)	Please analysis whether the given text contains complete medical events of the types #Imaging, Gene Testing, Operation, Medication#. If yes, please directly return the extracted event time, event type, event project name, and conclusion for the medical events belonging to #Imaging, Gene Testing, Operation, Medication # in JSON format: {"time": "","event_type": "","project": "","conclusion": ""}; otherwise, please return "No" The text to analysis is as follows: """*{text}*"""

A.3 Error Analysis on QLoRA-LLM Results

See Figure A3.

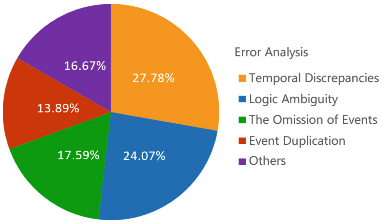


Fig. A3. The error analysis result on initial extracted results by QLoRA-LLM method. The total number of extracted medical events is 108 from 50 clinical notes.

References

1. Sun, W., et al.: Evaluating temporal relations in clinical text: 2012 i2b2 challenge. J. Am. Med. Inform. Assoc. **20**, 806–813 (2013)
2. Vaid, A., et al.: Using fine-tuned large language models to parse clinical notes in musculoskeletal pain disorders. Lancet Digit. Health **5**, e855–e858 (2023)
3. Martin, N., et al.: Recommendations for enhancing the usability and understandability of process mining in healthcare. Artif. Intell. Med. **109**, 101962 (2020)
4. Hassanpour, S., Langlotz, C.P.: Information extraction from multi-institutional radiology reports. Artif. Intell. Med. **66**, 29–39 (2016)
5. Tsujimura, T., et al.: Contextualized medication event extraction with striding NER and multi-turn QA. J. Biomed. Inform. **144**, 104416 (2023)
6. Ouyang, L., et al.: Training language models to follow instructions with human feedback. http://arxiv.org/abs/2203.02155 (2022)

7. Gero, Z., et al.: Self-Verification Improves Few-Shot Clinical Information Extraction. http://arxiv.org/abs/2306.00024 (2023)
8. Jimenez Gutierrez, B., et al.: Thinking about GPT-3 In-Context Learning for Biomedical IE? Think Again. In: Goldberg, Y., Kozareva, Z., and Zhang, Y. (eds.) Findings of the Association for Computational Linguistics: EMNLP 2022, pp. 4497–4512. Association for Computational Linguistics, Abu Dhabi (2022)
9. Dettmers, T., et al.: QLoRA: Efficient finetuning of quantized LLMs. http://arxiv.org/abs/2305.14314 (2023)
10. Zeng, A., et al.: GLM-130B: an open bilingual pre-trained model. http://arxiv.org/abs/2210.02414 (2023)
11. Liang, Y., et al.: RAAT: relation-augmented attention transformer for relation modeling in document-level event extraction. In: Carpuat, M., de Marneffe, M.-C., and Meza Ruiz, I.V. (eds.) Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 4985–4997. Association for Computational Linguistics, Seattle (2022)

Self-supervised Segment Contrastive Learning for Medical Document Representation

Waheed Ahmed Abro[1](✉), Hanane Kteich[2], and Zied Bouraoui[2]

[1] CDEP - UR 2471, Université d'Artois, Arras, France
wahmed.abro@univ-artois.fr
[2] CRIL - CNRS, Artois University, Arras, France

Abstract. Learning high-quality text embedding is vital for biomedical topic classification and many other NLP tasks. Contrastive learning has shown remarkable performance in generating high-quality text embeddings. However, existing methods typically generate anchor-positive pairs through discrete augmentations, simplifying the task of distinguishing positive from negative examples and limiting the learning of meaningful representations. In this paper, we present a self-supervised segment contrastive learning (SCL) approach designed for contrastively fine-tuning pre-trained language models. Our method randomly divides documents into anchor and positive segments, facilitating the learning of document embeddings by maximizing agreement between these segments. The proposed model contrastively fine-tune pre-trained ClinicalBioBERT language model to generate document embedding for medical documents. We evaluate our method on two publicly available medical datasets, MIMIC and Bioasq. Extensive experiments show that our proposed SCL approach outperforms baseline models, achieving superior performance in medical classification tasks.

Keywords: Document representation · Medical text · Contrastive learning · Language models

1 Introduction

Medical data processing plays a vital role in developing intelligent healthcare services. It involves various techniques for acquiring, managing, and analyzing health-related data. The biomedical literature is characterized by the continuous generation of textual data such as daily research articles, clinical notes, and healthcare summaries. Medical documents have quite different linguistic characteristics from normal documents, which can be a problem when classifying medical topics. It is important to develop a quality document embedding to encode these special medical entities. Document embedding is the process of encoding the documents into dense vector representation such that similar documents appear closer in the vector space. Document embedding plays an important role in classifying biomedical documents [9] and several other NLP tasks like information retrieval [2, 15], multiple-choice QA [12], etc.

Traditional document embedding approaches are based on fixed-length features such as bag-of-word and TF-IDF [19]. However, these methods fail to capture word order

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 312–321, 2024.
https://doi.org/10.1007/978-3-031-66538-7_31

and ignore semantic word distance, which limits their power on complex tasks. Recent advancements in transformer-based language models have led to a paradigm shift in document embedding, establishing them as the primary method for learning document representations from unlabelled corpora [4, 5, 13]. These models utilize vast amounts of unlabeled text data with a masked language modelling (MLM) objective to learn rich contextual sentence representations, which are then utilized by specific downstream tasks.

Despite their success, sentence representations generated by pre-trained language models suffer from anisotropy problems, where sentence representations occupy a narrow cone in the embedding space [6]. To alleviate this problem, contrastive learning [7, 28] has been proposed to bring similar sentences, i.e., anchor-positive pairs closer in the vector space, and dissimilar sentences, i.e., anchor-negative pairs far away in the vector space to uniform whole vector space.

The state-of-the-art models generate anchor-positive pairs by randomly augmenting the same sentence (e.g., Synonym replacement, delete one word, word repetition, dropout-noise), and anchor-negative pairs are chosen from the same mini-batch known as in batch-negative. Despite achieving success, employing discrete augmentations on sentences to generate positive pairs simplifies the task of distinguishing positive examples from negative ones; therefore, it does not lead to learning meaningful representations.

To address the above issues, we propose a self-supervised contrastive learning approach based on document-level objectives that can be used to contrastively fine-tune the pre-trained language model. Our proposed model randomly splits the document into two portions: a smaller text segment and a larger portion serving as an anchor text. The model learns document embeddings by training an encoder to maximize the agreement between anchor text and a positive text segment taken from the same document. As we are working with biomedical documents, we employed pre-trained ClinicalBioBERT [3] for contrastively finetuning.

Our primary contributions are:

- We propose an unsupervised self-supervised learning objective to contrastively fine-tune pre-trained ClinicalBioBERT model to induce high-quality document embeddings for medical documents.
- We conducted extensive experiments to highlight the advantages of learning document representation using proposed contrastive learning loss against state-of-the-art SimCSE and ESimCSE methods.
- We evaluated the quality of document embedding by training and evaluating multilayer perceptron (MLP) classifier on top of document embedding on three biomedical classification datasets.

2 Related Works

2.1 Medical Document Processing

Recently, transformer-based models are widely used for processing biomedical documents. To this end, [10] and [27] introduce sentence-aligned multilingual text simplification dataset for the medical domain, covering English, Spanish, French, Farsi, and Chinese. [11] and [1] introduce and investigate systems for generating medical reports from chest X-ray images. This includes integrating disease classification, transformer-based report generation, and an interpretation module to ensure consistency and clinical accuracy or evaluation metrics in text summarization and generation. [14] and [26] evaluated the capacity of LLM in medical domains, for clinical relation extraction or for medical systematic reviews, by adding medical knowledge into pre-trained models. [20] and [25] propose methods for medical prediction and automated medical report generation. [21] introduces MedCLIP, a framework for medical image-text contrastive learning that addresses challenges in pre-training on medical domain data. [17] survey discusses the application of pre-trained language models (PLMs) in the biomedical domain, highlighting their potential to improve performance on various natural language processing (NLP) tasks through pre-training on vast text corpora for universal language representation learning. In contrast, we propose to contrastive train these PLMs on the biomedical domain to generate high-quality document representation for medical documents.

2.2 Document Representation

Document representation learning employing self-supervised contrastive learning methods is a highly active research field [7, 18, 23, 24, 28]. In this direction, ConSERT [24] proposed to solve an anisotropy issue of BERT-derived sentence representation by contrastive training with adversarial attack, token shuffling, cutoff, and dropout augmentation methods. Similarly, SimCSE [7] uses a dropout version of the same sentence as a positive pair. In a similar context, ESimCSE [23] proposed word repetition as an augmentation method to generate positive pairs. On the other hand, SNCSE [18] applied negation to produce a negative pair. Similarly, DCLR [28] used noise-based negatives. Unlike the mentioned works, our work maximizes the agreement between the anchor and positive text segments taken from the same document to generate document representation.

3 Methodology

3.1 Self-supervised Contrastive Learning

We propose a Segment Contrastive Learning (SCL) framework to train a pretrained language model, i.e., ClinicalBioBERT and BERT-Base, using self-supervised contrastive learning. Our method maximizes the agreement between anchor text and a positive text segment taken from the same document. The model uses siamese network architecture to learn embeddings of text documents. We form a positive pair (a,s^+) by splitting the document into two portions: a smaller text segment (s^+) and a larger portion serving as an anchor text (a). Achor text and positive segment text are passed through the same encoder based on a pre-trained language model such as ClinicalBioBERT. The proposed method utilizes the [CLS] token representations to produce the anchor text (z_a) embedding and text segment (${z_s}^+$) embedding; then the classifier is trained to maximize the agreement between the segment and anchor text, both sampled from the same document and minimize the agreement between anchor text and segment taken from another document.

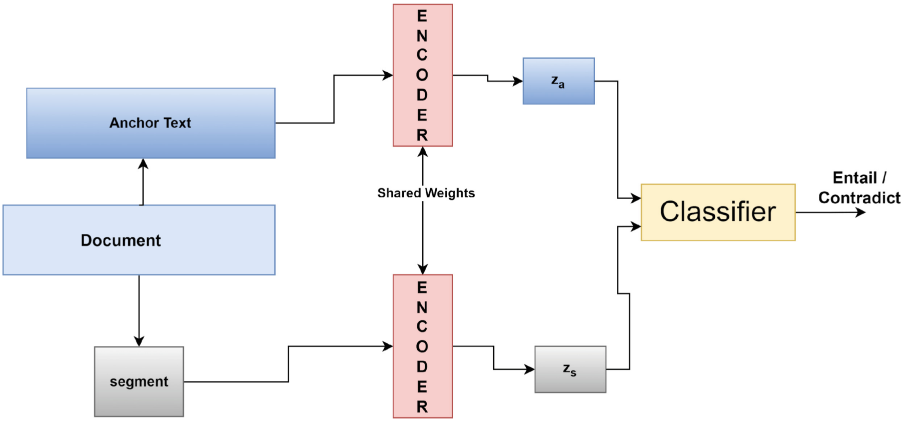


Fig. 1. Architecture of our proposed self-supervised learning objective. For each document in mini-batch, we construct a positive pair by partitioning it into two segments: a smaller text segment and a larger portion serving as an anchor text. Achor text and positive segment text are passed through the same encoder to generate embeddings z_a and z_s, respectively. The encoder is trained to minimize the distance between embeddings of text vector z_a and segment vector z_s of the same document and maximize the distance with segments of other documents that serve as negative samples (not shown here due to simplicity)

Suppose we have a mini-batch of N documents, denoted as $D = \{(d_i)\}^{N}{}_{1}$. For each document d_i, we randomly select a text segment s^+ and the remaining text as anchor text a to form a positive pair. We then select a negative segment s^- from the remaining $N - 1$ documents of the batch to serve as the negative pair,. One concern that could be seen here is the text segments could be similar and fit on the anchor text of several documents. However, this is not an issue as in the training objective, we have multiple negatives, so our model is forced to optimize most dissimilar documents than most similar ones. It is important to note that the segment predictive contrastive learning process can be viewed as an unsupervised natural language inference task, where a positive segment sample represents an entailment of a document, and negative samples from other documents represent a contradiction of the document. The multiple negatives ranking loss [8] function is used to optimize the model.

$$\mathcal{L} = -\frac{1}{n}\sum_{i=1}^{n} \frac{e^{(sim(z_{a_i}, z_{s_i}))}}{\sum_{1}^{k} e^{(sim(z_{a_i}, z_{s_k}^{-}))}} \qquad (1)$$

where z_{a_i} and $z_{s_i}^{+}$ are the vector representations of the anchor text segment and positive segment taken from the same document. The k negative sample of segments, vector representation is denoted by $z_{s_k}^{-}$, and sim is the cosine similarity. Multiple negative ranking loss compares the positive pair representation with the negative pair samples in mini-batch. The generalized architecture is shown in Fig. 1.

3.2 Document Classification

After producing robust document representation, we feed this representation as input to the classifier. The MLP classifier consists of three hidden layers and an output layer. We utilize the TANH activation function in the hidden layers and the sigmoid function in the output layer.

$$\hat{y} = \sigma(W * d + b) \tag{2}$$

where σ denotes the sigmoid activation function, d is the document representation, and W and b are the weights and biases of the hidden and output layers, respectively. Furthermore, the model minimizes the binary cross-entropy loss between true document labels and predicted document labels.

4 Experimental Setup

4.1 Datasets

We used two publicly available medical datasets namely, Medical Information Mart for Intensive Care (MIMIC-III) and Bioasq.

(MIMIC-III) [9] The MIMIC-III dataset comprises 50K discharge summaries from US hospitals, with each summary mapped to one or more (International Classification of Diseases, Ninth Revision) ICD-9 taxonomy labels. We utilized labels from the first level of the ICD-9 hierarchy. The dataset is partitioned into two categories: train and test. The training set encompasses 40000 summaries, while the test set consists of 10000, respectively. MIMIC-III is a multi-label classification dataset featuring 19 labels.

(BIOASQ) [16] The BIOASQ dataset comprises biomedical articles sourced from PubMed. Each article is annotated with concepts from the Medical Subject Headings (MeSH) taxonomy. We employed the 1st levels of the MeSH taxonomy. The dataset is divided into train and test categories. The training set contains 80000 summaries, and the test set consists of 20000, respectively. The BIOASQ is a multi-label classification dataset featuring 15 labels. Table 1 provides statistics of the datasets.

Table 1. Task type, number of classes, train and test examples statistics for each dataset

Dataset Task Type	No. Classes	Train	Test
MIMIC Muti-label Classification	19	40000	10000
Bioasq Muti-label Classification	15	80000	20000

4.2 Training Setup

We employ the ClinicalBioBERT[1] and Bert-Base model[2] pretrained-models. The proposed self-supervised SCL model is trained for 2 epochs with a batch size of 16 training

[1] https://huggingface.co/Dinithi/ClinicalBioBERT.

[2] https://huggingface.co/google-bert/bert-base-uncased.

samples. The length of the positive segments is set to 64 tokens. Adam optimizer with learning rates of $2e-5$ and weight decay of 0.001 is used to optimize the training loss. We used the transformers [22] library to train our model. The MLP models are trained for 16 epochs and finetuning models are trained for 4 epochs. We truncate and pad the document text to align with a maximum sequence input length of 512 tokens.

4.3 Baseline Models

To assess the quality of the representation of the document learned through the self-supervised method, we compare model performance against the pre-trained language model document Embedding, state-of-the-art contrastive learning of unsupervised sentence embedding models in feature-based and end-to-end finetuning settings:

– Embedding + MLP Classification: In this setting, we have frozen the parameters of the language model (ClinicalBioBERT and BERT) and applied an MLP classification layer for predicting the medical taxonomy labels.
– SimCSE [7] and ESimCSE Embedding [23] + MLP Classification: In this setting, we trained ClinicalBioBERT and BERT in a contrastive manner using the SimCSE and ESimCSE objectives. After training, we frieze the parameters and applied an MLP classification layer for predicting the medical taxonomy labels.
– BERT Classifier (Fine-tune): In the pre-trained BERT model [5], we employ a linear layer atop the final encoder layer's [CLS] token to classify medical taxonomy labels. The model is fine-tuned for 4 epochs with a batch size of 16.
– ClinicalBioBERT Classifier (Fine-tune): In the pre-trained ClinicalBioBERT model [3], we incorporate a linear layer on top of the last encoder layer [CLS] token for classifying medical taxonomy labels. The model is fine-tuned for 4 epochs with a batch size of 16.

4.4 Feature-Based Document Classification

Table 2 presents medical taxonomy classification results in the feature-based embedding setting where the ClinicalBioBERT and BERT encoder features are not updated during training. The table illustrates the results of various models across two distinct datasets, namely MIMIC and Bioasq, in terms of micro and macro F1 scores. The top rows display the performance of models utilizing ClinicalBioBERT Feature-based embedding. The ClinicalBioBERT Embedding$_{SCL}$ + MLP model produces the highest macro and micro F1 scores, achieving 54.65 and 71.28 macro F1-score on MIMIC and Bioasq datasets, respectively. This indicates that self-supervised *SCL* learning produces high-quality embeddings. Conversely, the state-of-the-art ClinicalBioBERT Embedding$_{SimCSE}$ + MLP and ClinicalBioBERT Embedding$_{ESimCSE}$ + MLP models does not enhance the performance of the baseline model Embedding + MLP. This observation suggests that relying solely on dropout augmentation or simple repetition of words to construct positive pairs and generate text embeddings may not yield significant improvements at the document or paragraph level embeddings. This stands in contrast to their demonstration of strong performance for sentence embeddings. Results demonstrate that the proposed SCL method improves embedding derived from ClinicalBioBERT and ClinicalBioBERT

Embedding$_{ESimCSE}$ by around 4% and 3% macro-F1 score on the MIMIC and Bioasq dataset, respectively.

Furthermore, shifting the focus to model's performance utilizing BERT-base Feature-based embeddings, we observe similar trends in performance. The BERT Embedding$_{SCL}$ + MLP model maintains its superiority with the highest macro F1 scores of 50.73 and 70.51 on MIMIC and Bioasq datasets, respectively. The proposed model outperforms all methods by producing approximately 8% and 3% better results in terms of macro-F1 on MIMIC and Bioasq datasets, respectively. Furthermore, results show that models trained on BERT Embedding models produce inferior results than models trained on ClinicalBioBert models. This indicates that training on domain-specific embeddings designed for biomedical text yields superior performance in medical classification tasks. Overall, the proposed SCL method generates high-quality embeddings which leads to model superior performance across different datasets and settings.

4.5 End-to-End Document Classification

To assess the quality of the embeddings learned by the proposed SCL model, we additionally perform end-to-end fine-tuning on the MIMIC and Bioasq datasets. The results are presented in Table 3. It is clear from the results that the SCL model outperforms other methods, achieving an improvement of approximately 1.2% and 1.6% in macro F1 score compared to the ClinicalBioBERT classifier and ClinicalBioBERT SimCSE classifier, respectively, on the MIMIC dataset. Additionally, there is an increase in performance of approximately 1% on the Bioasq dataset. It is apparent from the results that the BERT ESimCSE produces second-best results as it utilizes word repetition to de-bias the length for positive pairs.

Table 2. The proposed model performance against baseline methods using featurebased ClinicalBioBERT and BERT-base models on MIMIC and Bioasq datasets in terms of macro and micro F1-score.

	MIMIC		Bioasq	
	macro-F1	μ-F1	macro-F1	μ-F1
ClinicalBioBERT Embedding + MLP [3]	50.36	65.43	68.05	83.6
ClinicalBioBERT Embedding$_{SimCSE}$ + MLP [7]	48.01	62.61	69.13	83.31
ClinicalBioBERT Embedding$_{ESimCSE}$ + MLP [23]	50.85	64.42	68.77	83.17
ClinicalBioBERT Embedding$_{SCL}$ + MLP	54.65	66.41	71.28	84.43
BERT Embedding + MLP [5]	40.23	58.04	68.64	83.3
BERT Embedding$_{SimCSE}$ + MLP [7]	40.76	57.64	68.05	82.7
BERT Embedding$_{ESimCSE}$ + MLP [23]	42.02	58.09	68.04	82.66
BERT Embedding$_{SCL}$ + MLP	50.73	65.23	70.51	84.08

Furthermore, the final rows of Table 3 present the results of fine-tuning the pre-trained BERT-base model. It is evident from the results that fine-tuning the contrastively

Table 3. End-to-end classification performance of the proposed model against baseline methods using ClinicalBioBERT and BERT-base models on MIMIC and Bioasq datasets in terms of macro and micro F1-score.

	MIMIC		Bioasq	
	macro-F1	μ-F1	macro-F1	μ-F1
ClinicalBioBERT Classifier [3]	65.7	72.52	76.6	86.12
ClinicalBioBERT $_{SimCSE}$ Classifier [7]	65.38	70.89	76.62	85.99
ClinicalBioBERT $_{ESimCSE}$ Classifier [23]	66.35	72.64	76.97	86.32
ClinicalBioBERT $_{SCL}$ Classifier	66.92	72.68	77.35	86.64
BERT Classifier [5]	62.88	70.82	77	86.16
BERT$_{SimCSE}$ Classifier[7]	62.45	70.38	76.67	85.94
BERT $_{ESimCSE}$ Classifier [23]	64.04	70.96	76.84	85.91
BERT $_{SCL}$Classifier	64.77	71.24	77.49	86.26

pre-trained BERT model based on the SCL objective yields superior outcomes compared to the BERT classifier and BERT SimCSE Classifier on both the MIMIC and Bioasq datasets. This demonstrates that the contrastive learning task accelerates the fine-tuning process by learning high-quality document representations and facilitates the learning of a superior model.

5 Conclusion

We introduced a self-supervised segment contrastive learning (SCL) approach to learn document representation for medical documents. The proposed method randomly divides documents into anchor and positive segments, then contrastively fine-tune the pre-trained language models to maximize agreement between these segments. Through evaluations, we assessed the performance of the document embeddings against state-of-the-art baselines on different medical classification datasets, including MIMIC and Bioasq. Overall, our findings highlight the potential of SCL for refining pre-trained language models and addressing challenges in document representation learning. In future work, we plan to evaluate our method for complex information retrieval from biomedical documents.

Acknowledgments. This work was supported by ANR-22-CE23-0002 ERIANA and ANR Chaire IA Responsable.

References

1. Abacha, A.B., Yim, W.w., Michalopoulos, G., Lin, T.: An investigation of evaluation methods in automatic medical note generation. In: ACL, pp. 2575–2588 (2023)

2. Abro, W.A., Aicher, A., Rach, N., Ultes, S., Minker, W., Qi, G.: Natural language understanding for argumentative dialogue systems in the opinion building domain. Knowl.-Based Syst. **242**, 108318 (2022)
3. Alsentzer, E., et al.: Publicly available clinical BERT embeddings. arXiv preprint arXiv:1904.03323 (2019)
4. Clark, K., Luong, M.T., Le, Q.V., Manning, C.D.: ELECTRA: pre-training text encoders as discriminators rather than generators. arXiv preprint arXiv:2003.10555 (2020)
5. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. In: NAACL, pp. 4171–4186 (2019)
6. Ethayarajh, K.: How contextual are contextualized word representations? Comparing the geometry of BERT, ELMo, and GPT-2 embeddings. In: Inui, K., Jiang, J., Ng, V., Wan, X. (eds.) EMNLP-IJCNLP, pp. 55–65 (2019)
7. Gao, T., Yao, X., Chen, D.: SimCSE: simple contrastive learning of sentence embeddings. In: Moens, M.F., Huang, X., Specia, L., Yih, S.W.t. (eds.) EMNLP, pp. 6894–6910 (2021)
8. Henderson, M., et al.: Efficient natural language response suggestion for smart reply. arXiv preprint arXiv:1705.00652 (2017)
9. Johnson, A.E.W., et al.: MIMIC-III, a freely accessible critical care database. Sci. Data **3**, 1–9 (2016). https://api.semanticscholar.org/CorpusID:33285731
10. Joseph, S., et al.: Multilingual simplification of medical texts. arXiv preprint arXiv:2305.12532 (2023)
11. Nguyen, H.T., et al.: Automated generation of accurate & fluent medical x-ray reports. arXiv preprint arXiv:2108.12126 (2021)
12. Pang, R.Y., et al.: QuALITY: question answering with long input texts, yes!. In: Carpuat, M., de Marneffe, M.C., Meza Ruiz, I.V. (eds.) NAACL, pp. 5336–5358 (2022)
13. Radford, A., Narasimhan, K., Salimans, T., Sutskever, I., et al.: Improving language understanding by generative pre-training (2018)
14. Roy, A., Pan, S.: Incorporating medical knowledge in BERT for clinical relation extraction. In: EMNLP, pp. 5357–5366 (2021)
15. Sansone, C., Sperlí, G.: Legal information retrieval systems: state-of-the-art and open issues. Inf. Syst. **106**, 101967 (2022)
16. Tsatsaronis, G., et al.: An overview of the BIOASQ large-scale biomedical semantic indexing and question answering competition. BMC Bioinform. **16**(1), 1–28 (2015)
17. Wang, B., et al.: Pre-trained language models in biomedical domain: a systematic survey. ACM Comput. Surv.Surv. **56**(3), 1–52 (2023)
18. Wang, H., Li, Y., Huang, Z., Dou, Y., Kong, L., Shao, J.: SNCSE: contrastive learning for unsupervised sentence embedding with soft negative samples. CoRR abs/2201.05979 (2022)
19. Wang, S., Manning, C.: Baselines and bigrams: simple, good sentiment and topic classification. In: Li, H., Lin, C.Y., Osborne, M., Lee, G.G., Park, J.C. (eds.) ACL, pp. 90–94 (2012)
20. Wang, S., Liu, Z., Peng, B.: A self-training framework for automated medical report generation. In: EMNLP, pp. 16443–16449 (2023)
21. Wang, Z., Wu, Z., Agarwal, D., Sun, J.: MedCLIP: contrastive learning from unpaired medical images and text. arXiv preprint arXiv:2210.10163 (2022)
22. Wolf, T., et al..: Huggingface's transformers: Stateof-the-art natural language processing. ArXiv abs/1910.03771 arXiv:1910.03771 (2019)
23. Wu, X., Gao, C., Zang, L., Han, J., Wang, Z., Hu, S.: ESimCSE: enhanced sample building method for contrastive learning of unsupervised sentence embedding. In: Calzolari, N., et al. (eds.) COLING, pp. 3898–3907 (2022)
24. Yan, Y., Li, R., Wang, S., Zhang, F., Wu, W., Xu, W.: ConSERT: a contrastive framework for self-supervised sentence representation transfer. In: Zong, C., Xia, F., Li, W., Navigli, R. (eds.) ACL, pp. 5065–5075 (2021)

25. Yang, B., Wu, L.: How to leverage multimodal ehr data for better medical predictions? arXiv preprint arXiv:2110.15763 (2021)
26. Yun, H.S., Marshall, I.J., Trikalinos, T., Wallace, B.C.: Appraising the potential uses and harms of LLMs for medical systematic reviews. arXiv preprint arXiv:2305.11828 (2023)
27. Zhang, N., Zhang, Y., Guo, W., Mitra, P., Zhang, R.: FaMeSumm: investigating and improving faithfulness of medical summarization. arXiv preprint arXiv:2311.02271 (2023)
28. Zhou, K., Zhang, B., Zhao, X., Wen, J.R.: Debiased contrastive learning of unsupervised sentence representations. In: Muresan, S., Nakov, P., Villavicencio, A (eds.) ACL, pp. 6120–6130 (2022)

Sentence-Aligned Simplification of Biomedical Abstracts

Brian Ondov[1](✉) and Dina Demner-Fushman[2]

[1] Yale School of Medicine, 333 Cedar St, New Haven, CT 06510, USA
brian.ondov@yale.edu
[2] National Library of Medicine, Bethesda, MD, USA
ddemner@mail.nih.gov

Abstract. The availability of biomedical abstracts in online databases could improve health literacy and drive more informed choices. However, the technical language of these documents makes them inaccessible to healthcare consumers, causing disengagement, frustration and potential misuse. In this work we explore adapting foundation language models to the Plain Language Adaptation of Biomedical Abstracts benchmark. This task is challenging because it requires sentence-by-sentence simplifications, but entire abstracts must also be simplified cohesively. We present a sentence-wise autoregressive approach and report experiments with this technique in both zero-shot and fine-tuned settings, using both proprietary and open-source models. We also introduce a stochastic regularization technique to encourage recovery from source-copying during autoregressive inference. Our best-performing model achieves a 32 point increase in SARI and 6 point increase in BERTscore over the reported state-of-the-art. This also surpasses performance of recent open-domain and biomedical sentence simplification models on this task. Further, in manual evaluation, models achieve factual accuracy comparable to human-level, with simplicity close to that of humans. Abstracts simplified by these models could unlock a massive source of health information while retaining clear provenance for each statement to enhance trustworthiness.

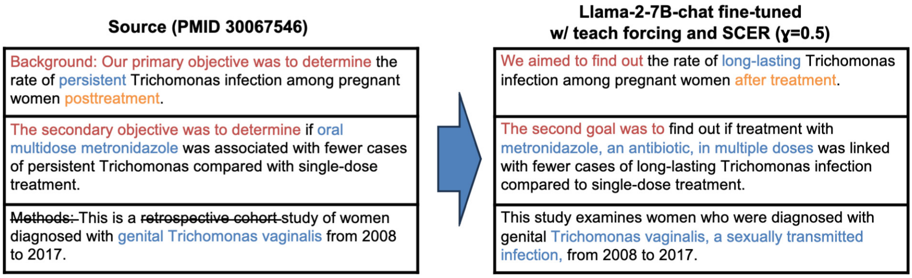
Keywords: Text simplification · Foundation Language Models · Biomedical literature

1 Introduction

The inability of patients to fully understand available information about their health has a significant impact on outcomes [6]. While many consumer-facing knowledge bases exist, these are cumbersome and labor-intensive to update and thus typically do not include the latest medical knowledge from the literature. When deeper questions are not answered by these resources, consumers may read beyond their expertise, potentially leading to misunderstanding [3,22].

Neural biomedical text simplification efforts to date have largely either framed the task as document-level plain language summarization [8,9,15] or

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 322–333, 2024.
https://doi.org/10.1007/978-3-031-66538-7_32



Source (PMID 30067546)

Background: Our primary objective was to determine the rate of persistent Trichomonas infection among pregnant women posttreatment.

The secondary objective was to determine if oral multidose metronidazole was associated with fewer cases of persistent Trichomonas compared with single-dose treatment.

~~Methods:~~ This is a ~~retrospective cohort~~ study of women diagnosed with genital Trichomonas vaginalis from 2008 to 2017.

Llama-2-7B-chat fine-tuned w/ teach forcing and SCER (γ=0.5)

We aimed to find out the rate of long-lasting Trichomonas infection among pregnant women after treatment.

The second goal was to find out if treatment with metronidazole, an antibiotic, in multiple doses was linked with fewer cases of long-lasting Trichomonas infection compared to single-dose treatment.

This study examines women who were diagnosed with genital Trichomonas vaginalis, a sexually transmitted infection, from 2008 to 2017.

Fig. 1. Excerpt of a PLABA abstract and output from our best-performing model. Notable changes are colored. Note that abstracts must be adapted sentence-wise, but as a whole, e.g. only explaining terms once.

simplification at the level of discrete sentences [12,19,20]. In contrast, the Plain Language Adaptation of Biomedical Abstracts (PLABA) [2] benchmark task requires sentence-aligned simplification of whole documents. This has the added challenge that each simplified sentence is affected by the context of the entire simplified abstract (Fig. 1). For example, added background or parenthetical explanations of terms will only occur the first time a term or concept is introduced, and whether an expert concept is explained or omitted depends on its centrality to the abstract. Further, anaphora may need to be resolved, and replacements for names of diseases, drugs, or study groups must remain consistent throughout the abstract for readers to follow them. As opposed to plain language summarization, which seeks to distill several takeaways, the sentence-aligned adaptation approach ensures more complete preservation of information that consumers might want, which may prevent them from circumventing the summary and going to the source. It also provides clear provenance for each statement, which is crucial for building trust, especially given the tendency of neural language models to confabulate.

In this work, we explore the use of foundation language models for the PLABA task, both in zero-shot and supervised fine-tuned settings. We evaluate models using automatic metrics on the PLABA test set and with manual judgments of simplicity, completeness, accuracy, and fluency. We find that both zero-shot GPT-3.5 and fine-tuned Llama 2 can generate simplifications with human-level factual accuracy even as they provide near-human levels of simplification. To our knowledge, our work is the first to include document context while performing sentence-wise simplification of biomedical documents. Our contributions are: (1) detailing methods for prompting and fine-tuning foundation language models to create sentence-aligned plain language adaptations of biomedical abstracts, (2) providing trained models for further research and use, and (3) extensive manual evaluations showing how models can simplify better and identifying where they are factually inaccurate. Code, model weights, outputs, and evaluations are available at https://github.com/ondovb/plaba-ft.

2 Methods

We explore several foundation language models: instruction fine-tuned GPT-3.5, Falcon [1], and Llama 2 [21]. In order to train models on a single GPU, we focus on model sizes with 13B parameters or fewer.

2.1 Sentence-Wise Autoregressive Prompting

The core of our method lies in progressively building prompts using system outputs as prior examples (Fig. 2). This takes advantage of the fact that foundation language models have typically been trained to be good in-context learners, following patterns in prompts and incorporating prior information [7]. In this approach, in an initial prompt, a general instruction is given (e.g. “Simplify:”) followed by the first source sentence prefixed with a label (e.g. “Original:”), and ending with a hanging label for completion (e.g. “Simple:”). The response is used to grow the prompt by filling in the first ‘simple’ sentence and providing the second sentence with the same labeling scheme. This continues until a response is obtained for each source sentence. Note that the responses can contain multiple sentences (essentially a split operation), but these can still be directly attributed to one source sentence.

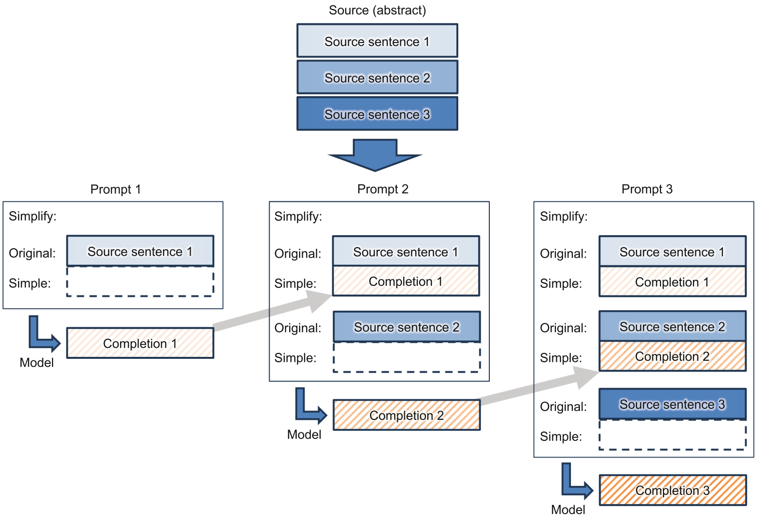


Fig. 2. Sentence-wise autoregressive prompting strategy. An initial prompt is provided with a general instruction (e.g. “Simplify:”), the first source sentence, labeled “Original:”, and the label “Simple:”. Subsequent prompts include all prior sentence/completion pairs, providing context while ensuring sentence alignment.

2.2 Supervised Fine-Tuning with Teaching Forcing

Though GPT-3.5 performs the task well with our prompting strategy, smaller, open-source models are desirable for many reasons, including, privacy, auditing, cost, and efficiency. Existing open-source foundation language models, however, lag far behind GPT-3.5 in zero-shot performance on this task. We thus sought to fine-tune such models. Since these models are purely causal, rather than sequence-to-sequence, supervised fine-tuning with gold outputs requires (1) constructing single inputs from training pairs, (2) inserting tags to mark the prompt and completion, and (3) masking tokens such that causal prediction and loss propagation is only performed for the section after the completion tag. Further, for efficient training, we use teacher forcing, as is common practice for autoregressive models. This means gold targets are used in training prompts where prior outputs would be inserted during inference (Fig. 6).

2.3 Source-Copying Exposure Regularization

A drawback of teacher forcing is exposure bias; i.e. a mismatch between prior generated outputs and the gold labels that were trained on, which may compound during autoregressive inference [5]. Further, pretrained language models are more likely to copy the source in machine translation settings [14]. In our case, even the gold training data contains targets that are similar or identical to the source, as annotators were instructed to leave simple language as-is. If a model leaves early sentences untouched during inference, in-context learning may be counterproductive, discouraging further sentences from being simplified. We thus introduce Source-Copying Exposure Regularization (SCER). For this method, rather than always using the gold label for teacher forcing, with some chance γ the source is copied instead (Fig. 3). We theorize this will gradually modify the model's in-context learning behavior so that it can still produce a simple output when appropriate, even when prior outputs seen in the prompt are similar or identical to sources.

2.4 Baselines

Other than the baselines presented with the PLABA dataset, we know of no published systems specifically designed for the PLABA task. We thus use recent state-of-the-art biomedical and open-domain sentence simplification models as additional baselines.

- T5-PLABA (Attal et al.) [2]: The best-performing baseline reported with the PLABA dataset.
- MUSS (Martin et al.) [16]: To our knowledge, the state of the art for open domain sentence simplification.
- BART-UL-ME (Flores et al.) [9]: Recent biomedical simplification method with strong results on several datasets. Since we require sentence-level outputs, we use the reported BART-XSum model fine-tuned on the Med-EASi corpus [4] (which is mostly single sentences) using Unlikelihood Loss, which was the best performing model for that dataset.

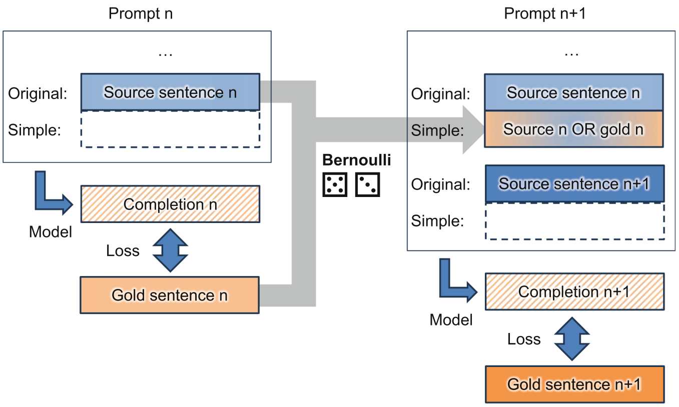


Fig. 3. Source-Copying Exposure Regularization (SCER). Rather than always using the gold standard for prior context, there is a chance that the source sentence is copied.

2.5 Implementation

All experiments were implemented in Python. For GPT-3.5, we use the OpenAI API. For fine-tuning open-source models, we use the HuggingFace transformers library [23] using Low-Rank Adapters (LoRA) [11], with $r = 16$ and $\alpha = 32$. A single NVidia A100 GPU was used for training and inference. All models had a batch size of 1 and maximum sequence length of 4,096. Llama-2-13B were 8-bit quantized. General instructions in prompts were "Rewrite for a lay audience:" for GPT-3.5 and "Simplify:" for open-source models. For SCER, we experiment with $\gamma \in \{0.25, 0.5, 0.75\}$.

3 Results

We evaluate GPT-3.5 zero-shot outputs in the following sections. Open-source models produce repetitive output with little simplification in the zero-shot setting but learn the task quickly with fine-tuning, with most needing only 100 examples for an initial large jump in validation BERTscore vs. zero-shot performance (Fig. 7). Continued training provides further improvement, with Llama 2 models performing better than Falcon, no benefit from larger Llama 2 models, and quicker training with SCER (Fig. 4). Note that 13B-parameter models have fewer steps because of longer computation time per step and equalized wall-clock training time. For SCER, all three values of γ have similar training trajectories; we choose $\gamma = 0.5$ for manual evaluation because it reaches the highest score.

3.1 Automatic Metrics

For automatic evaluation, we compare outputs on the PLABA test set to the gold standard adaptations using various relevant metrics, including BLEU [18], SARI [24], BERTscore [25], and Rouge [13]. Results are shown in Table 1.

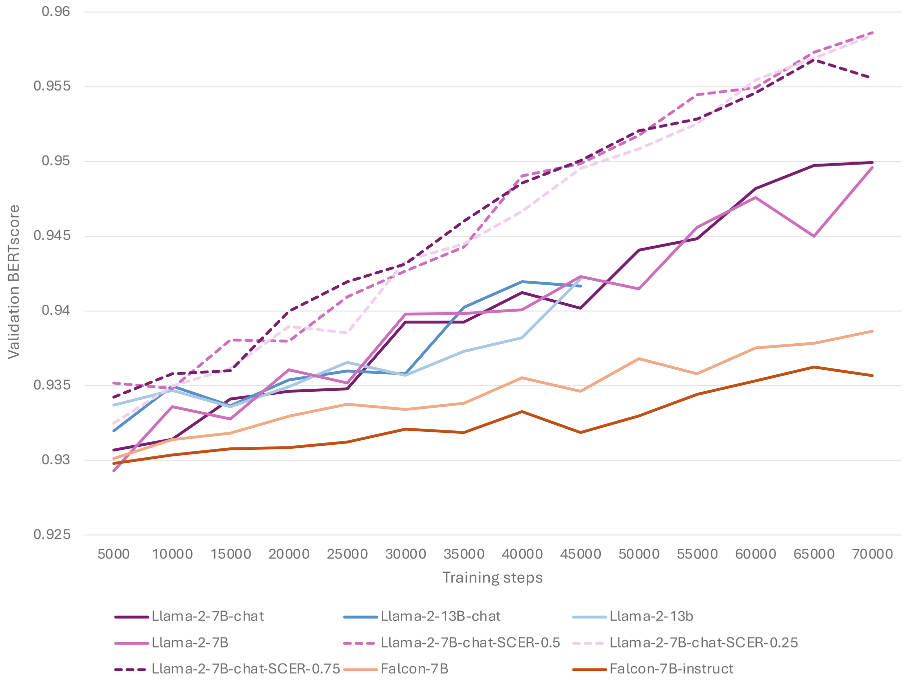


Fig. 4. BERTscores of checkpoints against the PLABA validation set.

Table 1. Performance of systems via automatic metrics on the PLABA test set.

Model	Rouge1	Rouge2	RougeL	BLEU	BERTscore	SARI
T5-PLABA[‡]	0.56	0.30	0.42	0.28	0.90	0.33
BART-UL-ME[†]	0.51	0.33	0.48	0.29	0.92	0.36
MUSS[†]	0.52	0.30	0.46	0.25	0.92	0.36
GPT-3.5*	0.45	0.20	0.37	0.17	0.92	0.34
Falcon-7B	0.65	0.48	0.62	0.45	0.94	0.49
Falcon-7B-instruct	0.65	0.49	0.63	0.45	0.94	0.47
Llama-2-7B	0.71	0.56	0.68	0.53	0.95	0.58
Llama-2-7B-chat	0.71	0.56	0.68	0.53	0.95	0.58
Llama-2-13B	0.66	0.50	0.64	0.46	0.94	0.48
Llama-2-13B-chat	0.67	0.50	0.64	0.47	0.94	0.48
Llama-2-7B-chat+SCER	**0.76**	**0.64**	**0.74**	**0.62**	**0.96**	**0.65**

*Zero-shot. [†]Cross-corpus. [‡]Previously reported results.

3.2 Manual Judgments

For manual evaluation, we chose an additional 40 abstracts using the same workflow as Attal et al. [2]. A pilot set of 3 abstracts was done by two annotators to compute inter-annotator agreement, which was generally high (Table 2); the rest were done by one annotator only. Following the four typical types of simplification judgments [17], each sentence was judged for *completeness*, *fluency*, *simplicity*, and *accuracy*, with simplicity and accuracy each judged at both sentence and term levels. Due to their more in-depth nature, sentence accuracy and completeness were performed for the three sentences of each abstract judged by both annotators to be most relevant to the consumer question. Judgments were performed on a 3-point likert scale and averages were linearly interpolated to a 0–100 scale. The two sub-axes (sentence and term) for both simplicity and accuracy were then averaged to create the final four axes. We manually evaluate (1) adaptations manually written by biomedical experts, as a human baseline, (2) GPT-3.5 zero-shot, for which automatic metrics are not a good measure, (3) the best-performing open-source model after fine-tuning (LLama-2-7B-chat), and (4) the latter with SCER, as an ablation experiment. Manual evaluations generally found simplifications to be of high quality (Fig. 5). The simplicity of Llama-2-7B-chat (78.80) increased to 83.53 with SCER, supporting the hypothesis that training specifically to recover from source copying prevents propagation of complex outputs through the autoregressive prompting process. All system outputs with the lowest judgment for factual accuracy (−1) can be seen in Table 3.

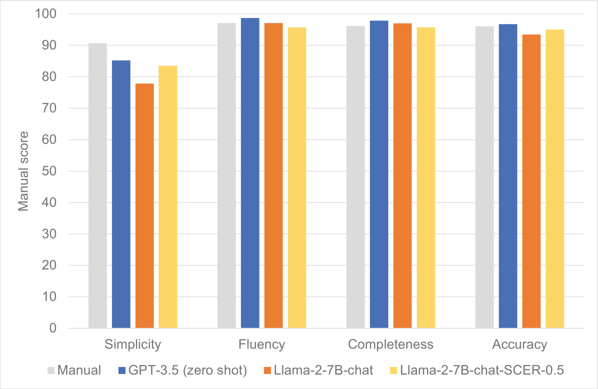


Fig. 5. Manual evaluation results for chosen systems on 40 additional abstracts.

3.3 Performance

During inference on the test set, Falcon-based models, on average, 26 s per abstract to create sentence-wise simplifications. Models based on Llama 2 took 31 s for 7B-parameter models and 68 s for 13B-parameter models.

4 Discussion

The models presented here show promise for making biomedical literature accessible to the general public. Yet, this work has several limitations. First, operating at the sentence level means that abstracts must be segmented first, which takes some time and is generally error-prone. Additionally, inference must be run n times for an abstract with n sentences. However, a benefit is that the entire original abstract is not needed as context to start generating output. It is thus not clear whether the strategy is costlier than a document-level approach, and more performance experiments are warranted. A further limitation is the relatively narrow scope of evaluation. The PLABA test set only contains 110 abstracts with 1,009 sentences, and manual evaluations only looked at 40 abstracts with 430 sentences due to the labor involved. As a proof-of-concept, and to test many variants, we fine-tuned open-source models for a relatively short amount of time (24 h) and did not yet see signs of overfit. Future studies are needed to explore the limits of training epochs and minimal dataset sizes. We also did not rigorously explore the effect of γ, the SCER source-copying probability. This value could be scheduled, perhaps reaching 1, similar to Bengio et al. [5], which would obviate autoregression and allow prompts with only the source sentences. Future work could also involve Reinforcement Learning from Human Feedback (RLHF), potentially using scores from our manual evaluations to train reward models.

5 Conclusion

In this work, we have shown that recent foundation language models are capable of simplifying biomedical abstracts sentence-by-sentence with factual accuracy similar to that of expert-written simplifications. Using a straightforward autoregressive prompting strategy, the proprietary GPT-3.5 model can perform this task zero-shot. While open source models, which may be desirable for both privacy and efficiency, lag far behind in the zero-shot setting, we show that they can be efficiently fine-tuned on a relatively small amount of data. This is enabled both by supervised fine-tuning with teacher forcing and a novel stochastic regularization regime that prevents degeneration into source-copying during inference. Both proprietary and fine-tuned models, however, fail to reach human levels of simplicity, according to manual evaluation. In closing this gap, we should continue to take care that accuracy is not compromised. Such a “progressive caution” approach [10] will allow incremental progress in simplicity while providing the benefits of current gains to consumers with minimal potential harm.

Acknowledgements. This work was supported by the NLM Intramural Research Program and utilized the computational resources of the HPC Biowulf cluster (http://hpc.nih.gov).

Appendix

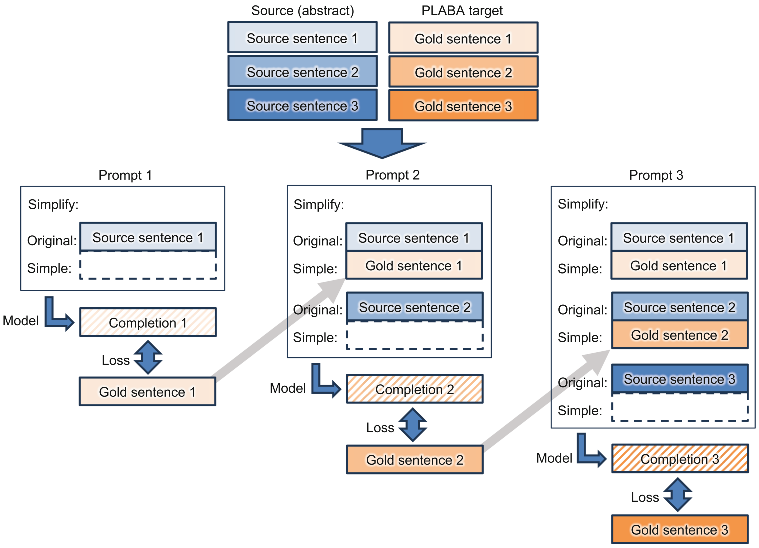


Fig. 6. Using teacher forcing to fine-tune for sentence-level autoregressive prompting. A 3-sentence abstract will be used to generate 3 different training prompts with gold standard completions used for both prior system outputs and as supervised labels. Loss is only propagated for the completion. Note that, during training, these prompts would be pooled with those from other abstracts and shuffled.

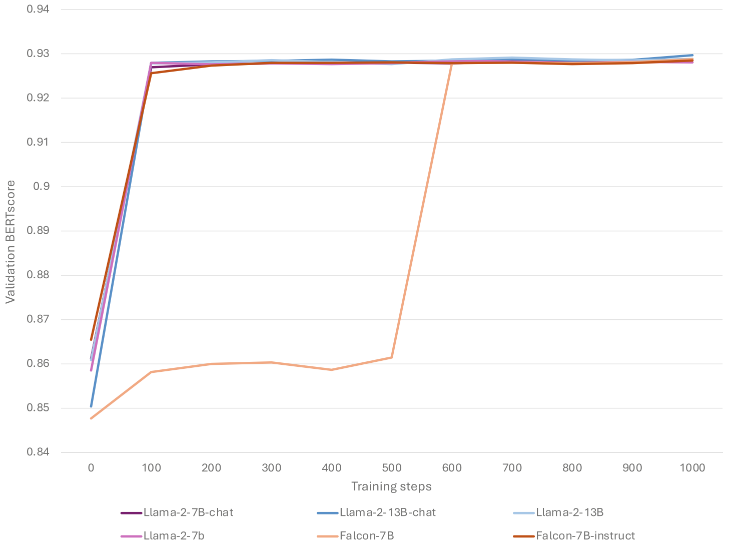


Fig. 7. BERTscores of checkpoints against the PLABA validation set for the first 1000 training steps.

Table 2. Inter-annotator agreement for the pilot manual evaluation, via Krippendorf's α, and at different levels of aggregation (sentence level being the original annotations). Agreement is generally high. Low agreement for fluency is due to over 98% of judgments being 1 (perfect), which breaks distributive assumptions of α. Sentence simplicity seems to be genuine disagreement, with one annotator judging mostly ones and the other with a more even mix.

Aggregation	Comp.	Flue.	Simp-term	Simp-sent	Acc-term	Acc-sent
Sentence	0.8833	−0.0255	0.5279	0.1748	0.5845	0.8148
Abstract	0.9684	0.3241	0.854	−0.1311	0.6757	0.8823
System	0.9853	0.0225	0.8077	0.0769	0.6026	0.7543

Table 3. System outputs manually judged to have poor accuracy (score of −1) at either the sentence or term level. Inaccurate sections of outputs are italicized.

GPT-3.5, PMID 25864273, sentence 9
Source: The disturbance of daily life due to NSP in postpartum women with past history of PMS and Hiesho were significant higher than that for women without those.
Output: Postpartum women who had a history of premenstrual syndrome (PMS) and *anemia* during pregnancy experienced more disruption in their daily lives due to neck and shoulder pain compared to women without these conditions.
GPT-3.5, PMID 7920896, sentence 3
Source: Laboratory evaluation revealed panhypopituitarism and pituitary cell antibodies were positive.
Output: The lab tests showed that she had a condition called panhypopituitarism, and antibodies were *found in her pituitary cells.*
Llama-2-7B-chat, PMID 7920896, sentence 4
Source: Both CT scans and MR images showed empty sella.
Output: Both brain scans showed the *pituitary gland was located in the bottom of the skull instead of in the middle of the brain.*
Llama-2-7B-chat, PMID 27935031, sentence 1
Source: Objectives: Interictal epileptiform discharges (IEDs) have been linked to memory impairment, but the spatial and temporal dynamics of this relationship remain elusive.
Output: Interictal epileptiform discharges (IEDs) are events *that can lead to seizures.* IEDs have been linked to memory impairment, but the spatial and temporal dynamics of this relationship remain elusive.
Llama-2-7B-chat+SCER, PMID 7920896, sentence 4
Source: Both CT scans and MR images showed empty sella.
Output: *T tests* showed she had empty *pituitary gland.*
Llama-2-7B-chat+SCER, PMID 32202633, sentence 6
Source: Exclusion criteria were qualitative studies; child age older than 12 years; and language assessment preverbal.
Output: Studies that were not quantitative (e.g. did not involve numerical data) were excluded. Additionally, studies on children older than 12 years and language assessment (measurement) of preverbal (*non-syllabic*) speech were excluded.

continued

Table 3. continued

Llama-2-7B-chat+SCER, PMID 33479851, sentence 9
Source: Other treatments in clinical development include fenfluramine in late phase III, perampanel, soticlestat-OV953/TAK-953, carisbamate and ganaxolone.
Output: Other treatments in clinical development include fenfluramine (*anti-depressant*), perampanel, soticlestat-OV953/TAK-953 (AED), carisbamate and ganaxolone (AEDs).
Llama-2-7B-chat+SCER, PMID 15359197, sentence 10
Source: At the three-cross cutoff point for ketonuria and at the 5 mmol/l cutoff point for ketonemia, the two tests had the same negative likelihood ratio (0.1), but the positive likelihood ratio of 3-beta-hydroxybutyrate (infinite) was higher than that of ketonuria.
Output: At a higher cutoff point for ketonuria and at the highest cutoff point for 3-beta-hydroxybutyrate, the two tests had the same *lack of accuracy* (0.1). However, the *accuracy* of 3-beta-hydroxybutyrate was higher than that of ketonuria (infinite).

References

1. Almazrouei, E., et al.: The falcon series of open language models. arXiv preprint arXiv:2311.16867 (2023)
2. Attal, K., Ondov, B., Demner-Fushman, D.: A dataset for plain language adaptation of biomedical abstracts. Sci. Data **10**(1), 8 (2023)
3. Aydın, G.Ö., Kaya, N., Turan, N.: The role of health literacy in access to online health information. Procedia Soc. Behav. Sci. **195**, 1683–1687 (2015)
4. Basu, C., Vasu, R., Yasunaga, M., Yang, Q.: Med-EASi: finely annotated dataset and models for controllable simplification of medical texts. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 37, pp. 14093–14101 (2023)
5. Bengio, S., Vinyals, O., Jaitly, N., Shazeer, N.: Scheduled sampling for sequence prediction with recurrent neural networks. Adv. Neural Inf. Process. Syst. **28** (2015)
6. Berkman, N.D., Sheridan, S.L., Donahue, K.E., Halpern, D.J., Crotty, K.: Low health literacy and health outcomes: an updated systematic review. Ann. Intern. Med. **155**(2), 97–107 (2011)
7. Brown, T., et al.: Language models are few-shot learners. Adv. Neural. Inf. Process. Syst. **33**, 1877–1901 (2020)
8. Devaraj, A., Marshall, I., Wallace, B.C., Li, J.J.: Paragraph-level simplification of medical texts. In: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 4972–4984 (2021)
9. Flores, L.J., Huang, H., Shi, K., Chheang, S., Cohan, A.: Medical text simplification: optimizing for readability with unlikelihood training and reranked beam search decoding. In: Findings of the Association for Computational Linguistics: EMNLP 2023, pp. 4859–4873 (2023)
10. Goodman, K.W., Miller, R.A.: Ethics in biomedical and health informatics: users, standards, and outcomes. In: Shortliffe, E.H., Cimino, J.J. (eds.) Biomedical Informatics, pp. 391–423. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-58721-5_12
11. Hu, E.J., et al.: LoRA: low-rank adaptation of large language models. arXiv preprint arXiv:2106.09685 (2021)

12. Kew, T., et al.: BLESS: benchmarking large language models on sentence simplification. In: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, pp. 13291–13309 (2023)
13. Lin, C.Y.: ROUGE: a package for automatic evaluation of summaries. In: Text Summarization Branches Out, pp. 74–81 (2004)
14. Liu, X., et al.: On the copying behaviors of pre-training for neural machine translation. In: Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pp. 4265–4275 (2021)
15. Lu, J., Li, J., Wallace, B.C., He, Y., Pergola, G.: NapSS: paragraph-level medical text simplification via narrative prompting and sentence-matching summarization. In: Findings of the Association for Computational Linguistics: EACL 2023, pp. 1079–1091 (2023)
16. Martin, L., Fan, A., De La Clergerie, É.V., Bordes, A., Sagot, B.: MUSS: multilingual unsupervised sentence simplification by mining paraphrases. In: Proceedings of the Thirteenth Language Resources and Evaluation Conference, pp. 1651–1664 (2022)
17. Ondov, B., Attal, K., Demner-Fushman, D.: A survey of automated methods for biomedical text simplification. J. Am. Med. Inform. Assoc. **29**(11), 1976–1988 (2022)
18. Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: BLEU: a method for automatic evaluation of machine translation. In: Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, pp. 311–318 (2002)
19. Pattisapu, N., Prabhu, N., Bhati, S., Varma, V.: Leveraging social media for medical text simplification. In: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 851–860 (2020)
20. Shardlow, M., Alva-Manchego, F.: Simple TICO-19: a dataset for joint translation and simplification of COVID-19 texts. In: Proceedings of the Thirteenth Language Resources and Evaluation Conference, pp. 3093–3102 (2022)
21. Touvron, H., et al.: Llama 2: Open foundation and fine-tuned chat models. arXiv preprint arXiv:2307.09288 (2023)
22. White, R.W., Horvitz, E.: Cyberchondria: studies of the escalation of medical concerns in web search. ACM Trans. Inf. Syst. (TOIS) **27**(4), 1–37 (2009)
23. Wolf, T., et al.: HuggingFace's transformers: state-of-the-art natural language processing. arXiv preprint arXiv:1910.03771 (2019)
24. Xu, W., Napoles, C., Pavlick, E., Chen, Q., Callison-Burch, C.: Optimizing statistical machine translation for text simplification. Trans. Assoc. Comput. Linguist. **4**, 401–415 (2016)
25. Zhang, T., Kishore, V., Wu, F., Weinberger, K.Q., Artzi, Y.: BERTScore: evaluating text generation with BERT. arXiv preprint arXiv:1904.09675 (2019)

Sequence-Model-Based Medication Extraction from Clinical Narratives in German

Vishakha Sharma[1](✉), Andreas Thalhammer[2], Amila Kugic[3], Stefan Schulz[3], and Markus Kreuzthaler[3]

[1] Roche Diagnostics, California, USA
vishakha.sharma@roche.com
[2] F. Hoffmann-La Roche AG, Basel, Switzerland
[3] Institute for Medical Informatics, Statistics and Documentation, Medical University of Graz, Graz, Austria

Abstract. Clinical narratives are a rich resource of patient-based information, where the automatic extraction of specific types of entities is still challenging due to various idiosyncrasies of non-standardized routine documentation. Much of the literature in clinical natural language processing (NLP) has been focused on English language. In this work, we focus on de-identified German-language clinical narratives. Predominant supervised approaches in the area of NLP, specifically named entity recognition (NER), need an expressive set of annotations for robust sequence modeling via a machine learning approach. This work investigates to what extent three main model types, conditional random fields (CRFs), bidirectional long short-term memory (BiLSTM) and Transformer models (BERT), perform on a limited set of annotations for medication information extraction in a specific clinical domain. The top performing feature, an optimized CRF model exploiting embedding representations out of a natively pre-trained language model using fastText, performed best with an overall F1 score 0.91. A multilingual BERT model incorporating German language resources reached an overall F1 score of 0.88, followed by the BiLSTM approach with an F1 score of 0.81. BERT based end-to-end systems nearly reach the performance of an optimized CRF approach. The results highlight the importance of pre-trained clinical language models available in languages other than English for fast, high performing, problem domain adaption.

1 Introduction

Electronic health records (EHRs), a main carrier of patient based information, can be seen as a multi-modal data source with different levels of structure and standardization. In that respect, semi-structured clinical documents are still a main carrier of information, where mentions and prescriptions of medications usually are documented in a narrative way [37]. Automatic recognition and

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 334–344, 2024.
https://doi.org/10.1007/978-3-031-66538-7_33

standardization of medication expressions is an essential task to support different research and hospital-based application scenarios, e.g., retrospective cohort building, adverse drug event detection or user-friendly EHR navigation and summarization [16], taking into account the idiosyncratic nature of clinical narrative data [19]. To support the generation of enhanced standardized and structured patient profiles, methodological enablers in the area of named entity recognition (NER) have evolved rapidly in the last years, yielding state-of-the art performance with transformer-based model architectures for natural language processing (NLP) tasks. The methodological shift to neural network based end-to-end systems [10] demands access to annotated data sources, which is often challenging in the clinical domain. Open clinical language resources in English are available in the Anglo-American region, for example with the MIMIC IV corpus [13], or via different scientific competitions (i2b2/n2c2 [11,35], ShARe/CLEF [8], SemEval [22]). For languages other than English, the current situation is less than ideal [21,32], with limited availability of clinical language resources such as German for the NLP community.

Specifically, the extraction of content of medical prescriptions into a structured template is still a challenging task and needs annotated language resources, so that model-based approaches can be supported. In this paper, we introduce our working hypothesis derived from a comprehensive review of state-of-the-art medication extraction systems, focusing particularly on sequence-model-based approaches.

1.1 Related Work

Methods and Tools. Alfattni et al. [2] investigated to what degree different embedding representation schemes influence the performance of medication information and relation extraction using the n2c2 annotated data set from 2018 [11]. The BiLSTM-CRF performed best in combination with pre-trained word and character embeddings, resulting in a lenient micro F1-measure of 0.92 for the NER task. The rule-based relation extraction approach, with a reported micro F1 score of 0.86, performed better than context-aware long short-term memories (LSTM). Kim and Meystre [15] used a stacked ensemble of CRFs, a search based prediction structure and recurrent neural network (RNNs), a BiLSTM network, for medication and related information extraction on the n2c2 data set with 505 annotated documents (303 training documents and 202 test documents). For relation classification, a Support Vector Machine (SVM) was trained. For medication information extraction, as well as for relation classification, an F1 score of 0.93 was reported. Tao et al. [33] applied CRFs, SVMs, Naive Bayes and Decision Trees for medication extraction using annotated data (medication name, dosage, mode, frequency, duration and reason) from the third i2b2 challenge on NLP for clinical records. For relation extraction, CRFs were used. The use of Glove embeddings [24] had a positive impact on the representation scheme for the extraction task, achieving an overall phrase level F1-measure of 0.86. Doan et al. [6] used the 2009 i2b2 data set to build and adapt a medication extraction pipeline by the use of existing tools (sentence boundary detection, SecTag [4], MedEx [38], Aspell, i2b2 output formatter) from the Vanderbilt University Medical Center. They reported an overall F1-measure of 0.82 for inexact matching. Meystre et al. [20] implemented the UIMA Textractor System for medication information detection. They tested their system using the 2009 i2b2 data set with an advanced pre-processing pipeline and leveraging MMTx [25] as a core medication extraction engine. The detection of medication routes performed best with an F1-measure of 0.86, prescription reason was extracted with an F1-measure of 0.29. MedEx [38], CLAMP [31], MedXN [30], and medExtractR [36] have to be mentioned in the context of medication extraction tool sets working on clinical narrative in English.

Scientific Competitions. In 2009, the third i2b2 workshop [35] focused on medications and medication-related information extraction. 20 teams participated in the challenge, with most of the top 10 teams tackling this challenge with rule-based approaches. The best performing [23] system, however, combined a rule and model-based (CRFs and SVMs) approach for the extraction task with a strict micro-average F1-measure of 0.86 and a F1-measure greater than 0.95 for the relationship classification task. The n2c2 series [11] of scientific competitions in the clinical NLP domain started in 2018. One outlined competition was the shared task on adverse drug events and medication extraction from EHRs. The top-performing team [39] used an ensemble of a CRF, a BiLSTM-CRF, and a BiLSTM-CRF topic-relation method. The top systems reached F1 scores 0.94 for concept extraction, 0.96 for relation classification, and 0.89 for end-to-end systems. The released data set consisted of 505 discharge summaries, 303 were used for training and 202 for testing the systems. In 2022, the n2c2 shared task [17] focused on medical event extractions and classification of the context of medication event descriptions. The aim was to extract medication descriptions, perform event classifications based on disposition, as well as process given change events to extract five contextual variables. The dataset consisted of 400 clinical notes for training and 100 for testing. For medication extraction with NER only, the top performing team reached an F1-measure of 0.97. Across all top performing teams, the most often applied methodology for medication information extraction was a NER approach that consisted of language model fine-tuning with sequence labeling.

Clinical Narratives in German. Caliskan et al. [3] utilized 10 annotated discharge letters for a first proof-of-concept for medication extraction evaluation (medication, dosage, mode, frequency, duration and reason), and applying the commercial NLP tool Averbis Health Discovery[1]. Medication-related phrases were detected with an F1-measure of 0.85, and for medication name detection an F1-measure of 0.94 was reported. Roller et al. [26] used Flair [1] to optimize various information extraction models and a convolutional neural network (CNN) on a relation extraction task leveraging an annotated nephrology corpus [27]. They

[1] https://averbis.com/health-discovery/.

reported an F1 score of 0.91 on 3,547 medication concepts. Frei and Kramer [7] machine-translated a corpus of 505 documents of the 2018 n2c2 challenge Track 2: Adverse Drug Events and Medication Extraction in EHRs [11] into German with the corresponding aligned annotation spans. The Spacy NER module was used with residual CNNs and Bloom embeddings [34]. Their NER model regarding medication information of the entities drug, strength, route, form, dosage, frequency and duration achieved an average F1 score of 0.82.

Based on this research overview, we want to explore whether classical model-based approaches for sequence labeling, e.g., CRFs, can compete with strong contextual models, e.g., LSTMs and transformer-based approaches in the clinical domain [40], by utilizing a small number of annotated data (factor 8 to 16 fewer annotations than the i2b2 and n2c2 data sets respectively). We hypothesize that CRFs, exploiting syntactical feature engineering, knowledge resources and embedding representation via pre-calculated language-specific models on a small set of labeled data, perform better than end-to-end LSTM and transformer-based approaches. Furthermore, we want to evaluate the impact of data augmentation for the CRF approach, as well as the impact of the use of a pre-trained language model on open domain web corpora versus a very language specific model trained on a small number of documents from the clinical language domain.

2 Materials and Methods

2.1 Data

Clinical Narratives. For the competitive method comparison, we used an excerpt of 1,696 dermatology discharge letters written in German extracted via Talend Open Studio from the clinical information system of KAGes, an Austrian network of public hospitals. The narratives on skin cancer were de-identified via the mandated medical data management team, and put onto a secure data lake, accessible for project members involved in the scientific investigation on-premises. The discharge letters exhibited numerous idiosyncrasies typical for non-standardized clinical routine documentation, such as abbreviations, acronyms, misspelling, domain-specific expert jargon, numeric expressions and German-specific, non-lexicalized single-word compounds.

Gold Standard. 500 randomly selected discharge letters were manually annotated by a trained fourth-year medical student, regarding medication-related information categorized by drug, strength, route and regimen, following academic competitions in the field of clinical NLP [11,35]. 400 documents (122,029 tokens) were used as a training set, 100 documents (32,477 tokens) as a test set. The brat rapid annotation tool was employed as a web-based annotation front end.

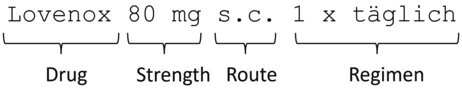


Fig. 1. Medication information entity types.

Drug. The entity type describes any brand name or drug substance mentioned in the narrative. "Lovenox" the brand name, see Fig. 1, with its active ingredient or drug substance name "Enoxaparin".

Strength. It is related to the prescribed drug dosage, as illustrated in Fig. 1. Even though strength values are often part of the brand name, in the annotation scheme the entity type strength is decoupled from the drug name proper, e.g., a full drug name description listed in the terminology resource ("LOVENOX 4.000 IE (40 mg)/0,4 ml Injektionslösung in einer Fertigspritze").

Route. Indicating the route of the drug administration, for example, from Fig. 1, "s.c." the abbreviation for "subcutaneous". Another typical example is "p.o." from Latin "per os" where the substance is administered through the mouth.

Regimen. In the competitions related to i2b2 and n2c2 also named 'Frequency', this entity type refers to how often a drug has to be taken over a certain period of time. Referring to Fig. 1 "1 x täglich" (once per day) is the corresponding entity.

Terminology Resources. As a standardized knowledge resource, we used the Austrian Register of Pharmaceutical Specialities - "Arzneispezialitätenregister" (ASP). It contains information about branded drug name, registration number, date of authorization, active ingredients and ATC (Anatomical Therapeutic Chemical) codes. The list was filtered for medications for humans, resulting in 16,571 entries. They were used for gazetteers and data augmentation for drug names with respect to their context.

Data Augmentation. The main goal of the augmentation step was to obtain various different examples of drug names and their local contexts. Therefore, all drug name entries in the ASP were put in the context line fetched out of the training set per document. Even though a drug name would not appear in a specific artificially generated context, we expected an overall more robust representation of the entity in the sequence modeling approach.

2.2 Methodological Approaches

Conditional Random Fields (CRF). We extended the Java-based CRF implementation of Mallet[2] [18] in order to handle embedding representations via fastText[3] [14] in combination with task specific feature engineering exploiting Gazetteers and syntactical features. We trained four different models resulting out of combinations of the following experimental parameters: a pre-trained German language model [9] based on open domain web-corpora (Lm_{op}), a German clinical domain-adapted language model built from scratch (Lm_{nt}), data augmentation exploiting a drug name dictionary generating all possible drug name occurrences in context (Aug_{y}) and no augmentation (Aug_{n}). We refer to this setting as CRF_{ext}.

Bidirectional Long Short-Term Memory (BiLSTM). Keras as a library encapsulates the implementation of a Long Short-Term Memory (LSTM) [12], featuring a bidirectional layer serving as a wrapper. This wrapper enhanced the model's ability to capture dependencies in both directions of the input sequence, making it more effective in tasks that require understanding the context for NER. The bidirectional layer that received input sequences with a length of 30, processed the input in both forward and backward directions, and produced an output with a feature dimension of 200. We used Adam for gradient optimization and trained the network for 20 epochs with a recurrent dropout rate of 0.1.

Bidirectional Encoder Representations from Transformers (BERT). The Simple Transformers library built upon the Hugging Face framework was chosen for the application of the BERT [5] method. It streamlined the process of training and evaluating downstream NER tasks. The base version of the BERT model was utilized, which had been pre-trained on multilingual data and retained case information. It was fine-tuned for the NER task with the German-language clinical narratives from our dataset. Our multilingual BERT model was trained with a maximum sequence length of 512 and with a training batch size of 8. We used Adam for gradient optimization and trained the network for 20 epochs with a learning rate of 2e-5.

Sequence Labeling. We attached to all three different core model types the well established BIO schema [28] at token level for the defined NER task.

3 Results and Discussion

Table 1 details strong performance of all models except for the entity type 'Route'. Given the limited availability of openly available medical NLP systems,

[2] https://mimno.github.io/Mallet/fst.html.
[3] https://fasttext.cc/.

especially for the German language, our approach highlights the potential to construct robust medical NER models. The tokenization of natural languages with complex structures can be challenging, and sequence models like BiLSTM and BERT might not always capture their linguistic characteristics in an optimal way. Additionally, these models are trained on large corpora containing text in many languages, resulting in a shared representation space. However, this means that these models have limited context for specific languages, e.g., German, therefore not yet reaching the performance of a feature-optimized and embedding representation enhanced CRF model, as shown in this investigation.

Table 1. Evaluation measurements by weighted macro-averaging per entity type using precision, recall and F1-measure [29] on the test set.

	CRF$_{ext}$				**BiLSTM**			**BERT**		
	Aug$_n$ Lm$_{op}$	Aug$_n$ Lm$_{nt}$	Aug$_y$ Lm$_{op}$	Aug$_y$ Lm$_{nt}$						
Entity	F1	F1	F1	F1	Precision	Recall	F1	Precision	Recall	F1
Drug	0.95	0.95	0.93	0.92	0.90	0.85	0.88	0.88	0.92	0.90
Strength	0.95	0.95	0.95	0.95	0.87	0.88	0.87	0.92	0.95	0.94
Route	0.55	0.54	0.54	0.54	0.38	0.45	0.41	0.48	0.67	0.56
Regimen	0.89	0.89	0.89	0.89	0.72	0.75	0.73	0.87	0.84	0.85
weighted avg	0.91	0.91	0.90	0.90	0.82	0.81	0.81	0.87	0.89	0.88

Error Analysis. The analysis contains the most noteworthy errors of the NER approaches, therefore gaining insights into possible systematic faults of misinterpreted contextual patterns. Interestingly the anatomical entity "Magen" (stomach) was frequently misclassified as a drug entity, furthermore, acronyms like "FDG" (F-18 Desoxyglucose) were often not recognized. For the entity type strength, "mg" was regularly classified at the beginning of a strength pattern, rather than defining the unit of an administrative drug subscription strength at the end. For the regimen entity type, patterns like "3 × 1" were often misclassified at the beginning of expressions like "bis zu 3 × 1 täglich" (up to three times per day). In contrast, the correct beginning of this regimen pattern was repeatedly missed, as well as occurrences of prescriptions, e.g., "4 × 1" (four times per day) have been a dominant source of error, e.g., "Ursofalk 4 × 1". Abbreviations in the form of "sc" (subcutaneous), "i.v." (intravenous), "p.os" (per os) were frequently not recognized for the administering route of a medication, therefore responsible for the moderate F1 score along all model types.

System Limitations. Compared to the most prominent competitions and data sets in the field, i2b2 and n2c2, the gold standard presented in this study has less defined entity types. This is caused by the fact that the most prominent entities under investigations were identified via a bottom up approach, neglecting those with very low occurrences. Having annotated about the same amount of documents as in the n2c2 challenge for medication extraction, the number of identified entities is less, therefore supporting our investigation on a small amount of annotated data. The de-identified documents were extracted out of one specialty, annotated by one expert and for the moment an investigation on the performance of relation classification has not been carried out yet.

4 Conclusion and Outlook

In this paper, we developed a system to extract medication information from German clinical narratives, focusing on the entity types drug, strength, route, and regimen. For this purpose, a gold standard of 500 de-identified narratives were annotated. Among the three sequence-model-based models under examination, the feature-enhanced CRF model, leveraging embedding representations from a natively pre-trained language model using fastText, demonstrated the highest performance, achieving an overall F1 score of 0.91 for the multi-class classification task. Following closely was a multilingual BERT model incorporating German language resources, which achieved an overall end-to-end F1 score of 0.88. In comparison, the BiLSTM approach trailed behind with an F1 score of 0.81. Notably, BERT-based end-to-end systems nearly matched the performance of the language optimized CRF approach. Utilizing a BERT-based medical language model in German like medBERT.de will be evaluated in the future but was not investigated at the moment due to scale-out possibilities of multilingual models on languages other than English.

We believe that the investigation presented here for extracting medication information holds potential to uncover trends and patterns associated with drug efficacy, adverse reactions, and real-world usage. This valuable data has the potential to guide research and development initiatives, facilitating the identification of opportunities for new drug development or enhancements to existing medications, and foster significant advancements in the future of German clinical NLP research. Global and local decision support systems, which take into account clinical narrative data, are in need for tailored NLP systems, where medication information is just one resource of interest for a fully structured and standardized patient profile. These NLP systems should leverage international standards like SNOMED CT and take into account other entities of interest, e.g., diagnostics, reasons for hospitalization, follow-up care instructions, allergies and vital signs. The need for openly available pre-trained domain-specific clinical language models in languages other than English are an indispensable part for the implementation of scalable and robust solutions in the future and are of interest for the global clinical NLP community.

Acknowledgements. This work was part of the Digital Biomarkers for Precision Medicine (DBM4PM) project within the K1 COMET Competence Center CBmed, funded by the Federal Ministry of Transport, Innovation and Technology (BMVIT);

the Federal Ministry of Science, Research and Economy (BMWFW); Land Steiermark (Department 12, Business and Innovation); the Styrian Business Promotion Agency (SFG); and the Vienna Business Agency. The COMET program is executed by the FFG. The work was mandated via the IRB (Ethics Committee) votes 32-194 ex 19/20 and 30-496 ex 17/18 from the Medical University of Graz. We acknowledge the financial support from Roche Diagnostics.

References

1. Akbik, A., Blythe, D., Vollgraf, R.: Contextual string embeddings for sequence labeling. In: Bender, E.M., Derczynski, L., Isabelle, P. (eds.) Proceedings of the 27th International Conference on Computational Linguistics, pp. 1638–1649. Association for Computational Linguistics, Santa Fe, New Mexico, USA (Aug 2018). https://aclanthology.org/C18-1139
2. Alfattni, G., Belousov, M., Peek, N., Nenadic, G.: Extracting drug names and associated attributes from discharge summaries: text mining study. JMIR Med. Inform. **9**(5), e24678 (2021)
3. Caliskan, D., et al.: First steps to evaluate an NLP tool's medication extraction accuracy from discharge letters. Stud. Health Technol. Inform. **278**, 224–230 (2021)
4. Denny, J.C., Spickard, A., 3rd., Johnson, K.B., Peterson, N.B., Peterson, J.F., Miller, R.A.: Evaluation of a method to identify and categorize section headers in clinical documents. J. Am. Med. Inform. Assoc. **16**(6), 806–815 (2009)
5. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
6. Doan, S., Bastarache, L., Klimkowski, S., Denny, J.C., Xu, H.: Integrating existing natural language processing tools for medication extraction from discharge summaries. J. Am. Med. Inform. Assoc. **17**(5), 528–531 (2010)
7. Frei, J., Kramer, F.: GERNERMED: an open German medical NER model. Softw. Impacts **11**, 100212 (2022)
8. Goeuriot, L., et al.: CLEF eHealth evaluation lab 2021. In: Hiemstra, D., Moens, M.-F., Mothe, J., Perego, R., Potthast, M., Sebastiani, F. (eds.) Advances in Information Retrieval: 43rd European Conference on IR Research, ECIR 2021, Virtual Event, March 28 – April 1, 2021, Proceedings, Part II, pp. 593–600. Springer International Publishing, Cham (2021). https://doi.org/10.1007/978-3-030-72240-1_69
9. Grave, E., Bojanowski, P., Gupta, P., Joulin, A., Mikolov, T.: Learning word vectors for 157 languages. arXiv preprint arXiv:1802.06893 (2018)
10. Hahn, U., Oleynik, M.: Medical information extraction in the age of deep learning. Yearb. Med. Inform. **29**(1), 208–220 (2020)
11. Henry, S., Buchan, K., Filannino, M., Stubbs, A., Uzuner, Ö.: 2018 n2c2 shared task on adverse drug events and medication extraction in electronic health records. J. Am. Med. Inform. Assoc. **27**(1), 3–12 (2020)
12. Hochreiter, S., Schmidhuber, J.: Long Short-Term Memory. Neural Computation **9**(8), 1735–1780 (11 1997). https://doi.org/10.1162/neco.1997.9.8.1735, https://doi.org/10.1162/neco.1997.9.8.1735
13. Johnson, A., Bulgarelli, L., Pollard, T., Horng, S., Celi, L.A., Mark, IV, R.: Mimic-iv (version 0.4). PhysioNet (2020)
14. Joulin, A., Grave, E., Bojanowski, P., Mikolov, T.: Bag of tricks for efficient text classification. arXiv preprint arXiv:1607.01759 (2016)

15. Kim, Y., Meystre, S.M.: Ensemble method-based extraction of medication and related information from clinical texts. J. Am. Med. Inform. Assoc. **27**(1), 31–38 (2020)
16. Kreuzthaler, M., Daumke, P., Schulz, S.: Semantic retrieval and navigation in clinical document collections. Stud. Health Technol. Inform. **212**, 9–14 (2015)
17. Mahajan, D., Liang, J.J., Tsou, C.H., Uzuner, Ö.: Overview of the 2022 n2c2 shared task on contextualized medication event extraction in clinical notes. J. Biomed. Inform. **144**, 104432 (2023). https://doi.org/10.1016/j.jbi.2023.104432, https://www.sciencedirect.com/science/article/pii/S1532046423001533
18. McCallum, A.K.: Mallet: A machine learning for language toolkit (2002) (2002)
19. Meystre, S.M., Savova, G.K., Kipper-Schuler, K.C., Hurdle, J.F.: Extracting information from textual documents in the electronic health record: a review of recent research. Yearb. Med. Inform. **17**(1), 128–144 (2008)
20. Meystre, S.M., Thibault, J., Shen, S., Hurdle, J.F., South, B.R.: Automatically detecting medications and the reason for their prescription in clinical narrative text documents. Stud. Health Technol. Inform. **160**(Pt 2), 944–948 (2010)
21. Névéol, A., Dalianis, H., Velupillai, S., Savova, G., Zweigenbaum, P.: Clinical natural language processing in languages other than English: opportunities and challenges. J. Biomed. Semantics **9**(1), 12 (2018)
22. Palmer, A., Schneider, N., Schluter, N., Emerson, G., Herbelot, A., Zhu, X. (eds.): Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021). Association for Computational Linguistics, Online (Aug 2021). https://aclanthology.org/2021.semeval-1.0
23. Patrick, J., Li, M.: High accuracy information extraction of medication information from clinical notes: 2009 i2b2 medication extraction challenge. J. Am. Med. Inform. Assoc. **17**(5), 524–527 (2010)
24. Pennington, J., Socher, R., Manning, C.D.: Glove: Global vectors for word representation. In: Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543 (2014). http://www.aclweb.org/anthology/D14-1162
25. Pratt, W., Yetisgen-Yildiz, M.: A study of biomedical concept identification: MetaMap vs. people. In: AMIA Annual Symposium Proceedings, pp. 529–533 (2003)
26. Roller, R., et al.: Information extraction models for German clinical text. In: 2020 IEEE International Conference on Healthcare Informatics (ICHI), pp. 1–2. IEEE (2020)
27. Roller, R., et al.: A fine-grained corpus annotation schema of German nephrology records. In: Proceedings of the Clinical Natural Language Processing Workshop (ClinicalNLP), pp. 69–77 (2016)
28. Sang, E.T.K., Buchholz, S.: Introduction to the conll-2000 shared task: Chunking. In: Proceedings of the Fourth Conference on Computational Natural Language Learning and of the Second Learning Language in Logic Workshop (CONLL/LLL 2000). Lissabon, Portugal, 13–14 september 2000, pp. 127–132. ACL (2000)
29. Schütze, H., Manning, C.D., Raghavan, P.: Introduction to information retrieval, vol. 39. Cambridge University Press Cambridge (2008)
30. Sohn, S., Clark, C., Halgrim, S.R., Murphy, S.P., Chute, C.G., Liu, H.: MedXN: an open source medication extraction and normalization tool for clinical text. J. Am. Med. Inform. Assoc. **21**(5), 858–865 (2014)
31. Soysal, E., et al.: CLAMP-a toolkit for efficiently building customized clinical natural language processing pipelines. J. Am. Med. Inform. Assoc. **25**(3), 331–336 (2018)

32. Starlinger, J., Kittner, M., Blankenstein, O., Leser, U.: How to improve information extraction from German medical records. IT - Information Technology **59**(4), 610 (Jan 2017)
33. Tao, C., Filannino, M., Uzuner, Ö.: Prescription extraction using CRFs and word embeddings. J. Biomed. Inform. **72**, 60–66 (2017)
34. Tito Svenstrup, D., Hansen, J., Winther, O.: Hash embeddings for efficient word representations. In: Advances in Neural Information Processing Systems **30** (2017)
35. Uzuner, Ö., Solti, I., Cadag, E.: Extracting medication information from clinical text. J. Am. Med. Inform. Assoc. **17**(5), 514–518 (2010)
36. Weeks, H.L., et al.: medExtractR: a targeted, customizable approach to medication extraction from electronic health records. J. Am. Med. Inform. Assoc. **27**(3), 407–418 (2020)
37. Xiao, C., Choi, E., Sun, J.: Opportunities and challenges in developing deep learning models using electronic health records data: a systematic review. J. Am. Med. Inform. Assoc. **25**(10), 1419–1428 (2018)
38. Xu, H., Stenner, S.P., Doan, S., Johnson, K.B., Waitman, L.R., Denny, J.C.: MedEx: a medication information extraction system for clinical narratives. J. Am. Med. Inform. Assoc. **17**(1), 19–24 (2010)
39. Xu, J., Lee, H.J., Ji, Z., Wang, J., Wei, Q., Xu, H.: UTH_CCB system for adverse drug reaction extraction from drug labels at TAC-ADR 2017. In: TAC. clamp.uth.edu (2017)
40. Yang, X., Bian, J., Hogan, W.R., Wu, Y.: Clinical concept extraction using transformers. J. Am. Med. Inform. Assoc. **27**(12), 1935–1942 (2020)

Social Media as a Sensor: Analyzing Twitter Data for Breast Cancer Medication Effects Using Natural Language Processing

Seibi Kobara(✉), Alireza Rafiei, Masoud Nateghi, Selen Bozkurt, Rishikesan Kamaleswaran, and Abeed Sarker

Emory University, 201 Dowman Dr, Atlanta, GA 30322, USA
seibi.kobara@emory.edu

Abstract. Breast cancer is a significant public health concern and is the leading cause of cancer-related deaths among women. Despite advances in breast cancer treatments, medication non-adherence remains a major problem. As electronic health records do not typically capture patient-reported outcomes that may reveal information about medication-related experiences, social media presents an attractive resource for enhancing our understanding of the patients' treatment experiences. In this paper, we developed natural language processing (NLP) based methodologies to study information posted by an automatically curated breast cancer cohort from social media. We employed a transformer-based classifier to identify breast cancer patients/survivors on X (Twitter) based on their self-reported information, and we collected longitudinal data from their profiles. We then designed a multi-layer rule-based model to develop a breast cancer therapy-associated side effect lexicon and detect patterns of medication usage and associated side effects among breast cancer patients. 1,454,637 posts were available from 583,962 unique users, of which 62,042 (10.6%) were detected as breast cancer members using our transformer-based model. 198 cohort members mentioned breast cancer medications, with tamoxifen as the most common. Our side effect lexicon identified well-known side effects of hormone and chemotherapy. Furthermore, it discovered a subjective feeling towards cancer and medications, which may suggest a pre-clinical phase of side effects or emotional distress. This analysis highlighted not only the utility of NLP techniques in unstructured social media data to identify self-reported breast cancer posts, medication usage patterns, and treatment side effects but also the richness of social data to answer such clinical questions.

Keyword: Breast cancer · natural language processing · social media

1 Introduction

Breast cancer, the most prevalent cancer among women, represents a significant public health concern. Accounting for about 30% of all new female cancer cases annually, it stands as the second leading cause of cancer death in women, following lung cancer [3, 8].

S. Kobara and A. Rafiei—Equally contributed to this paper.

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 345–354, 2024.
https://doi.org/10.1007/978-3-031-66538-7_34

Despite the grim statistics, there has been a consistent decrease in breast cancer mortality rates since 1989, with an overall decline of 43% through 2020 [4]. This notable progress is attributed to earlier diagnosis, increased awareness, and advancements in treatments [2, 9, 14]. However, the pace of this decline has shown signs of slowing in recent years [4], emphasizing the need for continued research and innovation in breast cancer care.

Despite the advances in breast cancer treatments, including endocrine therapy, which have led to declining death rates, as many as half to two-thirds of these breast cancer patients discontinue endocrine treatment within the first three years, increasing the risk of recurrence, hospitalization, and even death [7, 17]. Treatment non-adherence and discontinuation are often due to medication-related physical and mental side effects. Treatment-related side effects, or other subtle factors leading to non-adherence, are not detectable by laboratory diagnostic tests, but can be learned through patient communications. Information gleaned from patient communications (i.e., patient-reported outcomes (PROs)) are sometimes captured as free text in clinical narratives or through patient surveys. Both mechanisms of documenting information are labor-intensive and subject to biases. Furthermore, electronic health records (EHRs) have been found to under-document PROs (e.g., only 8% of a sample were found to contain PROs in a study) [7]. Thus, studies based solely on EHRs or other traditional instruments can only capture limited clinical information.

PROs or patient experiences are crucial in understanding the overall impact of breast cancer therapies and guiding future treatment strategies. One potential source of such information is social media, where patients are known to discuss their experiences with their peers. The potential of social media, specifically X (formerly known as Twitter), in this context is particularly compelling, because of its vast and diverse user base, and its ability to serve as a real-time global sensor for public sentiment and personal experiences. By tapping into the rich, unstructured data of social media, the trends and patient experiences that might remain hidden in clinical settings can be uncovered, tailoring for more patient-centered healthcare practices. However, obtaining information from real breast cancer survivors requires the establishment of a social media-based cohort, and then analyzing data posted by this cohort. Natural language processing (NLP) and machine learning methods provide potential solutions. Applying NLP techniques to data from X may offer insights into self-reported breast cancer-related information, medication use, and medication-related side effects across patients with various demographics, potentially surpassing the depth and breadth of traditional cohort studies. As such, the current study has been designed to:

- Identify self-disclosures of breast cancer from social media, build a cohort, and collect longitudinal data.
- Conduct NLP-driven analyses to detect and uncover patterns of medication usage among breast cancer patients and medication-associated side effects.
- Generate detailed statistics associated with the distribution of side effects observed across breast cancer-approved medications and identify potentially unknown medication and side effect associations.

2 Materials and Methods

2.1 Dataset

We collected a substantial dataset of 1,454,637 posts from 583,962 unique X users from March 22, 2021 to April 16, 2021. This dataset was compiled using four specific keywords: '*cancer* ', '*breastcancer*' (as a single term), '*tamoxifen*', and '*survivor* ', along with their hashtag equivalents (e.g., '*#breastcancer* '). An analysis of data collected using specific keywords revealed that although there were numerous health-related posts from genuine breast cancer patients, they were accompanied and often obscured by a significant amount of content posted by people who presented no evidence of being breast cancer patients/survivors (e.g., people sharing awareness about breast cancer). Four annotators in a study conducted by Al-Garadi et al. [1] processed and labeled a subset of 5,019 unique posts of this dataset into two classes: a) self, a family member, or friend-report of breast cancer (S), and b) not relevant posts (NR). The intuition behind this annotation was that if subscribers on X self-disclosed breast cancer statuses, those disclosures could be leveraged to create a social media-based breast cancer cohort. We have used this annotated data with the same train-test split for the supervised model development to extract the relevant posts from the full dataset for breast cancer medication and associated side effects analysis.

2.2 Self-reported Breast Cancer Post, Medication, and Side Effect Discovery

We adopt three distinct approaches to tackling the supervised classification of social media posts. Firstly, we extracted various feature sets from the text and constructed eight different classical machine learning classifiers. For this aim, we explored the combination of a broad spectrum of features, including n-grams (ranging from 1 to 3), word clusters [11], word-to-vector representations, text length, term frequency-inverse document frequency (TF-IDF), latent Dirichlet allocation (LDA) (i.e., extracted features based on latent topics of a post), sentiment score, and bidirectional encoder representations from transformers (BERT) embeddings, as the input features for training machine learning models. These models were then optimized through a grid search method, involving an extensive range of parameters and 5-fold cross-validation on the training dataset. Additionally, we developed a two-layer BLSTM model (parameters: unit = 100, dropout = 0.2, recurrent dropout = 0.2), followed by a dense layer (parameters: unit = 100, dropout = 0.2). We also fine-tuned the pre-trained transformer-based architectures and weights of the BERT and BERT large models on the available training dataset.

We created two lexicons manually from the annotated text to represent medication expressions and their associated side effects. The medication lexicons were built upon both drug and commercial names approved by the Food and Drug Administration (FDA) in the available National Cancer Institute (NCI) medication library[1]. Similarly, side effect lexicons were built upon the side effects listed by the NCI[2] as well as COVID-19 symptoms [12] lexicons.

[1] https://www.cancer.gov/about-cancer/treatment/drugs/breast. [Accessed 02-19-2024].

[2] https://www.cancer.gov/about-cancer/treatment/side-effects. [Accessed02-19-2024].

The posts were collapsed by unique usernames as we observed that the time span of the available posts was one month, by which we assumed that breast cancer medication prescription patterns and side effects are constant in this particular time span. Then, multiple independent annotators manually annotated these collapsed posts, and the newly found medications and side effects were added to the lexicons. To assess and monitor the performance of the rule-based models over the enrichment of the lexicons, we considered random annotated usernames as the gold standard set. Figure 1 illustrates the designed workflow for the multi-layer rule-based model development to extract breast cancer medications and their associated side effects. The multi-layer rule-based model is composed of two distinct models, each tasked with identifying medications and side effects separately, employing the concept of inexact matching. Notably, Levenshtein string similarity was used for recognition, accommodating near-misspellings and paraphrased expressions. The models utilized a rolling sliding window, ranging from 1 to 9, with a stride of one, to capture both single-word and multi-word entities. Of note, the model was engineered to prevent the redundant detection of the same words using different window sizes. They also incorporated a feature for negation detection using a list of negation triggers. If a negation is detected, it is flagged accordingly in the final result. Our approach was particularly focused on precision (at the expense of recall) based on the fact that there is no shortage of data from a social media-based cohort, and, consequently, avoiding false positives is more important than avoiding false negatives.

2.3 Descriptive Analysis of Medications and Side Effects

We first defined a breast cancer cohort using the best-performing classifier and applied our rule-based models to identify mentions of breast cancer medications and side effects. To describe the distribution of side effects, we classified breast cancer-approved medications using biological mechanism-based functional classification, specifically hormone therapy, chemotherapy, immune checkpoint inhibitors, and kinase inhibitors. Recognizing that a single medication could be mentioned multiple times in aggregated posts based on unique members in the cohort, we considered only unique occurrences of medications or side effects per sample. As a cancer treatment regimen, in general, may consist of several medications, several functional classifications of medication appeared for each cohort member. Therefore, we first identified the patterns of medications and then tested the association between these medication patterns and side effects using the Kruskal-Wallis test. Multiple tests across side effects were adjusted using Benjamini Hochberg correction. Pair-wise comparisons across medication patterns were performed using Dunn's test.

The scripts used in this study were implemented using Python (version: 3.8.8) and R (version: 4.3.2) and are available on GitHub[3]. The level of significance was set to 0.05.

[3] https://github.com/seibikobara/NLPBreastCancerSideEffectDiscovery.

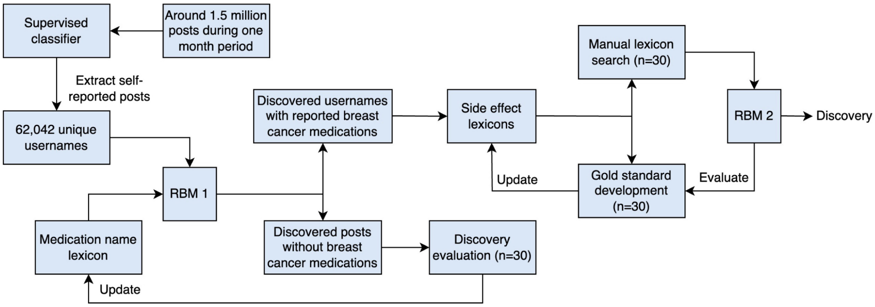

Supervised classifier
Around 1.5 million posts during one month period
Extract self-reported posts
62,042 unique usernames
RBM 1
Medication name lexicon
Update
Discovered usernames with reported breast cancer medications
Discovered posts without breast cancer medications
Side effect lexicons
Discovery evaluation (n=30)
Update
Manual lexicon search (n=30)
Gold standard development (n=30)
RBM 2
Discovery
Evaluate

Fig. 1. Flow diagram of the methods for medication and their associated side effects discovery from the social media cohort data. "n" represents the number of users, and "RBM" stands for rule-based model.

3 Results

3.1 Self-reported Breast Cancer Post, Medication, and Side Effect Discovery

Among the developed supervised classifiers, the transformer-based BERT-large language model achieved the highest performance with an accuracy of 0.93 and an F_1 score of 0.89 on the test data. As such, we used this model as the supervised classifier in the proposed workflow. The model was trained with a maximum sequence length of 100 and batch size of 16 during 40 epochs. During the rule-based model development, the manual annotation was done by three independent annotators, and the pair-wise inter-annotator agreements were calculated using the Cohen's Kappa measure [15]. In the initial annotation round, the average agreement was 0.78. The annotators discussed the disagreements until full agreement was reached. The multi-layer rule-base model successfully identified breast cancer-approved medications and associated side effects with an *F1* score of 0.64, precision of 0.64, and recall of 0.64.

3.2 Descriptive Analysis of Medications and Side Effects

We detected multiple medication mentions in the discovered breast cancer posts collapsed by unique usernames, remaining 62,042 users with 10.6% prevalence in the available users in X. In this cohort, 198 members expressed a minimum of one breast cancer-approved medication. Many cohort members mentioned taking medications without specifying their names, and we excluded all such cases. 109 (55.1%) mentioned tamoxifen, and hormone therapy was the most expressed medication category in the posts. Figure 2 presents the full distribution of medication mentions. 31 side effects were identified, and the most commonly expressed side effect was body ache & pain (34 [17.2%]) (Fig. 3). In the discovered breast cancer cohort, seven patterns of medication patterns were identified including hormone therapy, chemotherapy, a combination of hormone therapy and chemotherapy, a combination of hormone therapy and kinase inhibitor, immune checkpoint inhibitor, a combination of hormone therapy, kinase inhibitor, and immune checkpoint inhibitor, and a combination of hormone therapy and immune checkpoint inhibitor. A Kruskal-Wallis test showed that 17 out of 31 side effects

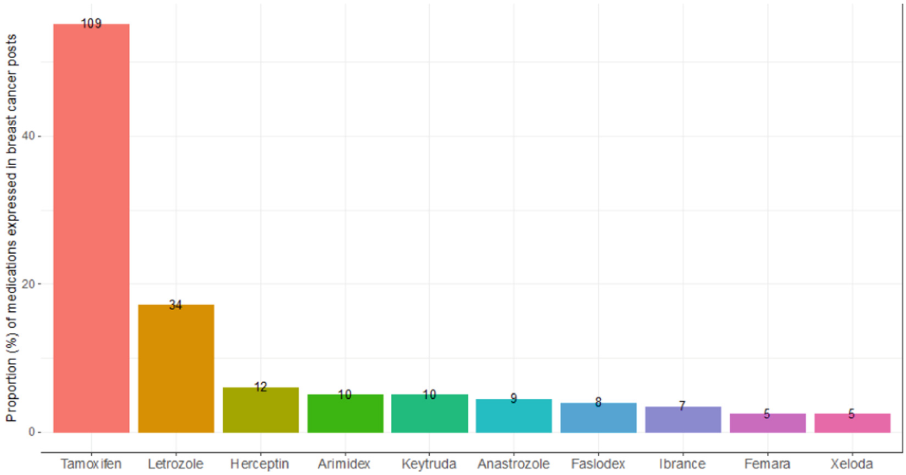


Fig. 2. Top 10 most expressed breast cancer approved medications in our social media cohort. The label on top of the bar charts represents the number of cohort members who expressed medications.

were significantly associated with the medication patterns (adjusted p-value <0.05), of which known side effects of hormone or chemotherapy include pyrexia, body ache, anxiety, nerve problems, and hair loss. In addition, our novel breast cancer-associated side effect lexicon discovered a generalized side effect or negative emotion, not elsewhere classified (NEC), which represents a subjective feeling towards cancer and medications, such as '*worst feeling*' or '*feeling of dreadful side effect*' (Fig. 4). The prevalence of the generalized side effect or negative emotion, NEC in a combination of hormone therapy and chemotherapy was significantly higher than the prevalence in hormone therapy (adjusted p-value <0.05).

4 Discussion

We trained a transformer-based model to identify self-reported breast cancer posts on social media and create a cohort. The developed supervised transformer-based classifier demonstrates superior performance compared to the classical machine learning methods that worked with the extracted features, thanks to its innovative architecture and pretraining scheme. This finding highlights the feasibility of constructing a large breast cancer cohort from social using an automated NLP pipeline and detecting breast cancer therapy-associated side effects using lexicon development. The methods may also be replicated to create other similar cohorts.

Using a multi-layer rule-based model architecture that was optimized for precision, we detected medication name expressions and side effects in each unique cohort member profile. In our analysis of accounts discussing breast cancer, hormone therapy was the most expressed medication category, with tamoxifen being the most commonly mentioned keyword. By developing a novel breast cancer therapy-associated side effect lexicon, we identified patterns of side effects that were related to medication patterns.

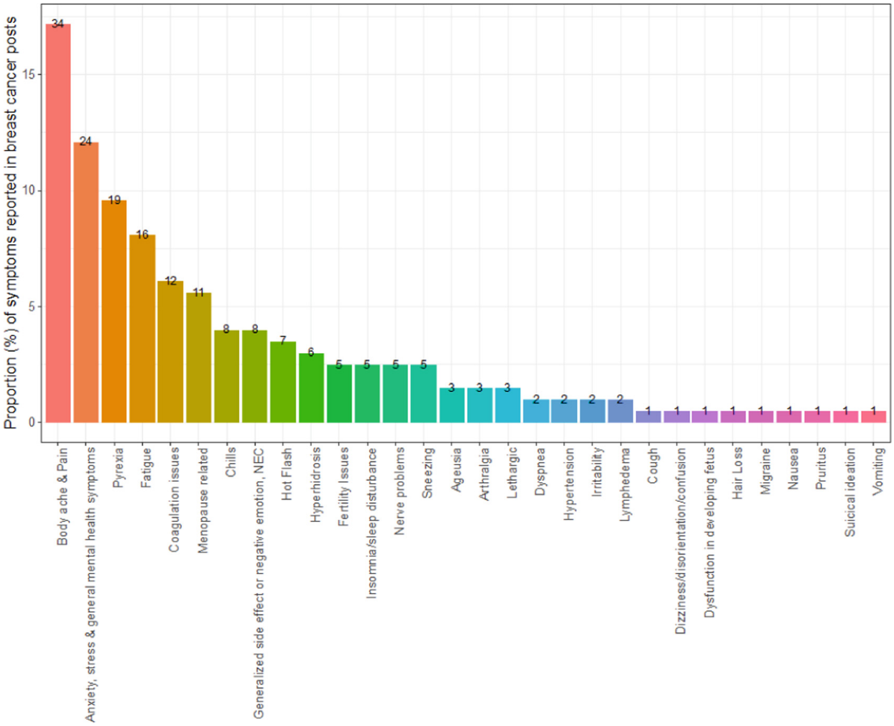


Fig. 3. Expressed side effects in our social medial cohort. The y-axis represents the proportions and the text labels on top of bar charts are the number of users who expressed side effects. "NEC" stands for not elsewhere classified.

We discovered that breast cancer therapy is associated with a broad range of side effects, as expressed by the cohort members. Chemotherapy is associated with a number of neurological side effects, including nausea, pain, and hair loss [10]. Kinase inhibitors for breast cancer are associated with adverse and side effects in the cardiovascular system, such as hypertension, atrial fibrillation, and heart failure, gastrointestinal, and skin reactions [5, 6, 13]. Clinical trials of immune checkpoint inhibitors indicated that colitis and pneumonitis are the most frequent fatal adverse effects of immune checkpoint inhibitors [16]. Notably, we were able to identify well-known side effects of hormone therapy and chemotherapy, such as pyrexia, body ache, anxiety, and nerve problems. Furthermore, our novel breast cancer therapy-associated side effect lexicon discovered the generalized side effect or emotion, NEC. Although this lacks a detailed description of side effects, our cohort users may suffer from an indescribable feeling, which may be a pre-clinical side effect of emotional distress or common side effects of breast cancer therapy.

Our work demonstrates the utility of a social media-based cohort that is created automatically via NLP and machine learning methods for identifying patterns of medications and side effects. Such cohorts, once the methods are established and deployed, can grow automatically over time, leading to the collection of seemingly unlimited data.

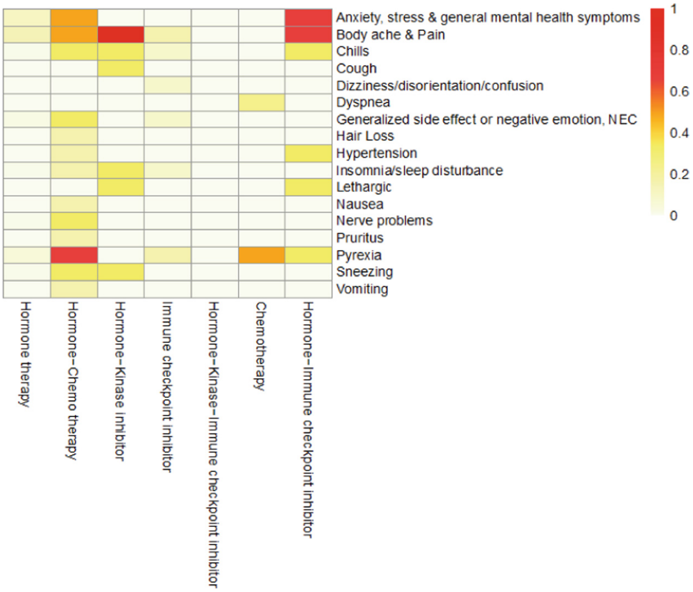


Fig. 4. The heat map of prevalence of significantly associated side effects with medication patterns (adjusted p-value < 0.05). NEC, not elsewhere classified.

Potentially novel insights may then be mined using the strategies we described in this paper. The discovered side effects using our novel breast cancer-associated side effect lexicon represent potential hypotheses that can be studied and validated through more traditional studies. Such methods of cohort data analyses may be particularly useful for new medications entering the market for which post-marketing surveillance data is limited or absent. Such strategies may also enable the early detection of potential unknown side effects. Also, while the side effects discussed on social media may not be severe (e.g., nausea), they may be the reasons for non-adherence among patients, an association that needs to be investigated in future research.

Several aspects of the analysis should be carefully considered for future improvement. First, although X is a widely used text-based social media compared with other platforms, posts in X often lacked enough context for accurate classification. This limitation leads to potential misidentification of breast cancer-related posts, posing challenges in estimating cancer prevalence or detecting specific cancer types. Second, we collapsed the posts based on accounts, assuming the homogeneity of the medications' prescription regimens and associated side effects. Multiple medication expressions appearing in one of the cohort members' profiles can lead to false positive indications of the association between medication and its side effects. Lastly, a limited sample size of members who mentioned medications may induce selection bias and underpower our lexicon to discover a side effect occurrence.

5 Conclusion

A supervised classifier was able to identify a self-reported breast cancer cohort. Multiple rounds of lexicon development of medications and side effects were conducted, and rule-based models were designed to describe the medication usage prevalence and their links to side effects. We demonstrate, for the first time, the feasibility of an NLP model discovering the patterns of side effects associated with breast cancer-approved medications. Notably, our breast cancer therapy-associated side effect lexicon identified a potential pre-clinical side effect in breast cancer therapy. The next steps include investigating the proposed workflow in a larger sample size and other social media platforms. This can involve the consideration of the usage of non-breast cancer medications and assessing the magnitude of side effects alleviation due to supportive medications.

References

1. Al-Garadi, M.A., et al.: Automatic breast cancer cohort detection from social media for studying factors affecting patient-centered outcomes. In: Michalowski, M., Moskovitch, R. (eds.) AIME 2020. LNCS, vol. 12299, pp. 100–110. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-59137-3_10
2. Berry, D.A., et al.: Effect of screening and adjuvant therapy on mortality from breast cancer. N. Engl. J. Med. **353**(17), 1784–1792 (2005)
3. Giaquinto, A.N., Miller, K.D., Tossas, K.Y., Winn, R.A., Jemal, A., Siegel, R.L.: Cancer statistics for African American/black people 2022. CA Cancer J. Clin. **72**(3), 202–229 (2022)
4. Giaquinto, A.N., et al.: Breast cancer statistics, 2022. CA Cancer J. Clin. **72**(6), 524–541 (2022)
5. Iancu, G., et al.: Tyrosine kinase inhibitors in breast cancer. Exp. Ther. Med. **23**(2), 1–10 (2022)
6. Le Du, F., Diéras, V., Curigliano, G.: The role of tyrosine kinase inhibitors in the treatment of HER2+ metastatic breast cancer. Eur. J. Cancer **154**, 175–189 (2021)
7. McCowan, C., et al.: Cohort study examining tamoxifen adherence and its relationship to mortality in women with breast cancer. Br. J. Cancer **99**(11), 1763–1768 (2008)
8. Miller, K.D., et al.: Cancer statistics for the US Hispanic/Latino population, 2021. CA Cancer J. Clin. **71**(6), 466–487 (2021)
9. Munoz, D., et al.: Effects of screening and systemic adjuvant therapy on ER-specific US breast cancer mortality. J. Natl. Cancer Inst. **106**(11) (2014)
10. Pearce, A., et al.: Incidence and severity of self-reported chemotherapy side effects in routine care: a prospective cohort study. PLoS ONE **12**(10), e0184360 (2017)
11. Sarker, A., Gonzalez, G.: A corpus for mining drug-related knowledge from twitter chatter: language models and their utilities. Data Brief **10**, 122–131 (2017)
12. Sarker, A., Lakamana, S., Hogg-Bremer, W., Xie, A., Al-Garadi, M.A., Yang, Y.C.: Self-reported covid-19 symptoms on Twitter: an analysis and a research resource. J. Am. Med. Inform. Assoc. **27**(8), 1310–1315 (2020)
13. Shyam Sunder, S., Sharma, U.C., Pokharel, S.: Adverse effects of tyrosine kinase inhibitors in cancer therapy: pathophysiology, mechanisms and clinical management. Signal Transduct. Target. Ther. **8**(1), 262 (2023)
14. Tong, C.W.S., Wu, M., Cho, W.C.S., To, K.K.W.: Recent advances in the treatment of breast cancer. Front. Oncol. **8**, 227 (2018)

15. Viera, A.J., Garrett, J.M., et al.: Understanding interobserver agreement: the kappa statistic. Fam. Med. **37**(5), 360–363 (2005)
16. Wang, D.Y., et al.: Fatal toxic effects associated with immune checkpoint inhibitors: a systematic review and meta-analysis. JAMA Oncol. **4**(12), 1721–1728 (2018)
17. Waterhouse, D.M., Calzone, K.A., Mele, C., Brenner, D.E.: Adherence to oral tamoxifen: a comparison of patient self-report, pill counts, and microelectronic monitoring. J. Clin. Oncol.Oncol. **11**(6), 1189–1197 (1993)

Bioinformatics and Omics

Breast Cancer Subtype Prediction Model Integrating Domain Adaptation with Semi-supervised Learning on DNA Methylation Profiles

Joung Min Choi and Liqing Zhang(✉)

Virginia Tech, Blacksburg, VA 24061, USA
lqzhang@cs.vt.edu

Abstract. Breast cancer is a highly heterogeneous disease, leading to the varied drug resistance and clinical outcomes. Accurate identification of breast cancer subtypes is crucial for precise diagnosis, treatment decision-making, and prognosis prediction. Recent research has highlighted the significant role of epigenetic alterations in breast cancer development, particularly the potential of aberrant DNA methylation patterns as subtype-specific markers. However, challenges exist in developing a breast cancer subtype prediction model based on DNA methylation profiles, primarily due to the limited number of available samples with subtype information.

In this study, we propose BCtypeFinder, a breast cancer subtype prediction framework utilizing a domain adaptation network with semi-supervised learning. Our model leverages both labeled and unlabeled DNA methylation datasets to learn domain-invariant features, aligning the distributions of the same breast cancer subtypes across different datasets. BCtypeFinder outperforms existing methods, demonstrating superior classification performance in several scenarios. We also investigated the effectiveness of batch correction in BCtypeFinder, revealing its capability to eliminate batch distinctions among patients with the same subtype across different batches, thus enhancing the classifier's robustness. BCtypeFinder is publicly accessible at https://github.com/joungmin-choi/BCtypeFinder.

Keywords: Breast cancer subtype prediction · DNA methylation · Domain adaptation · Semi-supervised learning

1 Introduction

Breast cancer is the second most prevalent cancer and the leading cause of cancer-related deaths among women globally [1]. The heterogeneity of breast cancer arises from dynamic variations in molecular components throughout tumor progression, manifesting differences at transcriptomic, epigenetic, and genomic levels [2]. This heterogeneity, both inter-tumor and intra-tumor, contributes to diverse drug resistance patterns and clinical outcomes, presenting a challenge in predicting prognosis and therapy responses for

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 357–366, 2024.
https://doi.org/10.1007/978-3-031-66538-7_35

breast cancer patients [3]. To enhance diagnostic precision and effectiveness of targeted medicine, efforts have been made to classify breast cancer into five molecular intrinsic subtypes using signature genes associated with hormone receptors, proliferation, myoepithelial, and basal features (PAM50) [4]. This standardized breast cancer subtyping system plays pivotal roles in cancer prognostication and therapeutic decision-making [5].

In recent years, extensive research has delved into epigenetic alterations contributing to breast tumorigenesis, recognizing them as pivotal drivers in cancer development and the transition from normal tissue to neoplasia and metastasis [6]. The hypermethylation of CpG promoters in breast cancer cells leads to the silencing of numerous well-known tumor suppressor genes [7]. Conversely, hypomethylation is identified as an early event in tumorigenesis, serving as an indicator of tumor progression and prognosis [8]. In the current landscape of therapeutic approaches, there is a shift towards targeting epigenetic alterations rather than genetic mutations, driven by the potential reversibility of epigenetic changes [9].

Epigenetic analyses have revealed that aberrant DNA methylation patterns are associated with the molecular subtypes of breast cancer, suggesting their potential as subtype-specific markers [10]. However, a major challenge in developing a cancer subtype classifier using DNA methylation profiles stems from the scarcity of methylome datasets with subtype label annotations, leading to overfitting during model training. To address this issue, a cancer subtype classification framework called meth-SemiCancer has been introduced, leveraging semi-supervised learning (SSL) [11]. Through SSL, meth-SemiCancer utilizes unlabeled methylation datasets by assigning pseudo-labels based on the model's predictions, which are then fed back into the model for retraining, ultimately enhancing generalization. Meth-SemiCancer has demonstrated improved subtype classification performance across various cancers, including breast cancer, highlighting the potential of SSL in mitigating overfitting issues during model training. However, meth-SemiCancer does not consider possible batch effect due to the diverse unlabeled datasets generated by different labs.

In the realm of aligning different datasets and jointly training classifiers, domain adaptation has been introduced in the computer vision field [12]. This approach leverages information learned from a source domain with an adequately labeled dataset to enhance model performance on a different yet related target domain containing unlabeled datasets. Recently, domain adaptation has found widespread application in cell type inference and classification using single-cell and spatial transcriptomic data [13, 14]. Treating each batch as a domain, single-cell transcriptomics datasets are integrated to alleviate variations caused by batch effects and distribution discrepancies, enhancing model generalization and robustness against local perturbations or noise. However, these methodologies have primarily been assessed in settings where the target domain possesses a single set of unlabeled data and the source domain encompasses a substantial number of samples, frequently surpassing 30,000 samples. This presents a challenge when attempting to adapt them to DNA methylation profiles, which typically contain fewer than 1,000 samples per dataset.

The present work proposes BCtypeFinder, a breast cancer subtype prediction model that leverages domain adaptation networks through semi-supervised learning of DNA methylation profiles. The feature extractor and cancer subtype classifier modules are initially trained using a source dataset with cancer subtype labels. Subsequently, adversarial training is employed to extract domain-invariant features, which are then fine-tuned in a semi-supervised learning phase with subtype alignment for batch correction. BCtypeFinder shows superior performance over the start-of-the-art model meth-SemiCancer and other machine learning-based classifiers. Further ablation studies and visualization of features extracted from BCtypeFinder affirm its efficacy in smoothing batch effects in diverse methylation datasets and enhancing classifier robustness.

2 Methods

Let n_s and n_t denote the number of samples in the source and target data, respectively, m the number of CpGs common to all the data, $X_s = \left(x_1^s, \ldots, x_{n_s}^s\right) \in R^{n_s \times m}$ the source DNA methylation data matrix with cancer subtype labels, and $X_t = \left(x_1^t, \ldots, x_{n_t}^t\right) \in R^{n_t \times m}$ the target methylation data matrix representing multiple unlabeled datasets. In the context of domain adaptation, X_s and X_t are assumed to be different yet related. BCtypeFinder comprises feature extractor, domain discriminator, and subtype classifier modules, undergoing three training phases: (1) Pre-training, (2) Adversarial training, and (3) Fine-tuning based on semi-supervised learning and subtype alignment (Fig. 1).

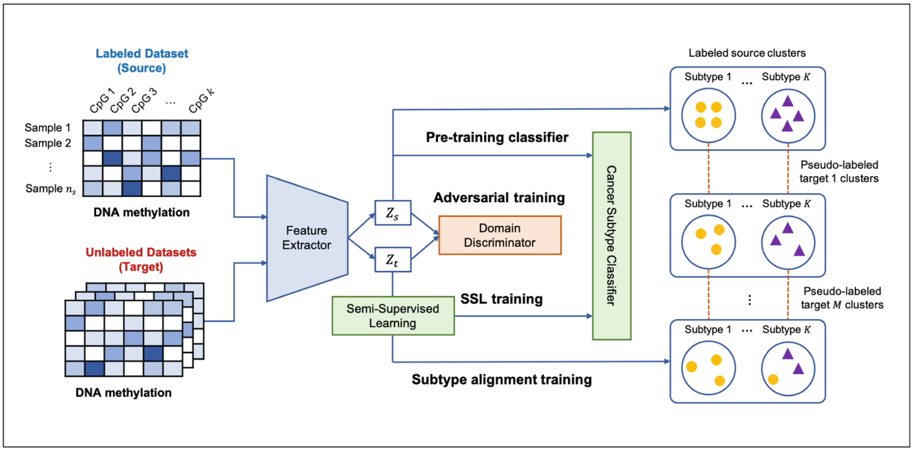


Fig. 1. Illustration of the proposed breast cancer subtype classification model, BCtypeFinder.

BCtypeFinder initiates its training by pre-training on the source dataset X_s, which includes annotated subtype labels. The objective is to initialize the weights in the feature extractor and the subtype classifier. Both modules consist of two fully connected layers, with the hidden layer in the classifier followed by a softmax function to estimate the posterior probability of each breast cancer subtype. The training minimizes the subtype classification error using cross-entropy loss:

$$\mathcal{L}_{PT} = -\frac{1}{n_s}\sum\nolimits_{i=1}^{n_s}\sum\nolimits_{j=1}^{K} y_{i,j}\log\left(p_{i,j}\right), \tag{1}$$

where K is the number of subtypes, $y_{i,j}$ the binary indicator for whether subtype label j is correct for sample i, and $p_{i,j}$ the predicted probability of sample i belonging to the subtype j.

Following pre-training, adversarial training is employed for both the feature extractor and the domain discriminator composed of two hidden layers and a softmax function. These modules engage in a competitive learning process to acquire domain-invariant features. The adversarial domain adaptation loss is minimized, with the domain discriminator D trained to distinguish the origin dataset of the extracted features, while the feature extractor F strives to transfer the target dataset's distribution to the source, confusing the domain discriminator by maximizing the loss:

$$\min_{D}\max_{F}\mathcal{L}_{\mathrm{AT}} = -\frac{1}{n_s + n_t}\sum\nolimits_{i=1}^{n_s+n_t}\sum\nolimits_{j=1}^{M} y_{i,j}\log(D(F(x_i))), \tag{2}$$

where M is the number of domains (i.e., datasets with different batch effects), and $y_{i,j}$ and $D(F(x_i))$ are the actual and the model predicted domain probability of the sample i, respectively.

To enhance the generalization of the subtype classification model and refine predictions, fine-tuning is executed through semi-supervised learning (SSL) with subtype alignment. Pseudo-labels for the unlabeled target dataset are obtained by assigning the breast cancer subtype with the highest posterior probability, continually updated during each iteration. The optimization of the feature extractor and subtype classifier modules relies on the weighted cross-entropy loss for both the source and target datasets:

$$\mathcal{L}_{SSL} = -\frac{1}{n_s}\sum\nolimits_{i=1}^{n_s}\sum\nolimits_{j=1}^{K} y_{i,j}\log\left(p_{i,j}\right) - \alpha(t)\frac{1}{n_t}\sum\nolimits_{i=1}^{n_t}\sum\nolimits_{j=1}^{K} y'_{i,j}\log\left(p_{i,j}\right), \tag{3}$$

where K is the number of subtypes, $y_{i,j}$ and $y'_{i,j}$ the true and pseudo-subtype probability distribution for the labeled and unlabeled datasets, respectively, and $p_{i,j}$ the model-predicted probability of sample i belonging to the j-th subtype. The coefficient $\alpha(t)$ is introduced to balance the classification training loss between the source and target datasets, gradually increasing to prevent poor local minima in the optimization process,

$$\alpha(t) = \begin{cases} 0, & t < T_1 \\ \frac{t-T_1}{T_2-T_1}\alpha_f, & T_1 \le t < T_2 \\ \alpha_f, & T_2 \le t \end{cases} \tag{4}$$

where t is current epoch, $T_1 = 100$, and $T_2 = 200$.

After a few epochs of SSL training to assign stable pseudo-labels to each sample in target datasets, subtype alignment is performed iteratively with SSL training. As a batch correction approach, the extracted features belonging to the same subtype should be well-clustered, and those from different domains should be mapped nearby. To achieve this, samples from each dataset in different domains are grouped based on the subtype, and the centroid for each subtype is calculated and explicitly aligned. The centroid is defined as the mean embedding of each subtype, with pseudo-labels obtained each iteration for centroid calculation in the target dataset. During training, the distance between the center of all centroids for each subtype and each domain's corresponding centroid is minimized:

$$\mathcal{L}_{SA} = -\frac{1}{K}\frac{1}{M}\sum\nolimits_{k=1}^{K}\sum\nolimits_{m=1}^{M}|C^k - C_m^k|_2, \tag{5}$$

where C^k represents the centroid of subtype k in the source data and C_m^k is the centroid of subtype k in the domain m data.

3 Experimental Design

3.1 Data Collection and Preprocessing

The DNA methylome breast cancer dataset obtained from TCGA [15] (referred to as TCGA-BRCA) was used as the source. It comprises 1,060 solid primary breast tumor tissue samples with methylome measured by Illumina Human Infinium 450K and 27K assays and subtype information obtained from [4]. Subtypes include LumA, LumB, Her2, Basal, and Normal-like. Three publicly available breast cancer datasets (GSE69914 [16], GSE75067 [17], and GSE72245 [18]) containing 611 samples obtained from the Gene Expression Omnibus (GEO), were used as the unlabeled target data. Detailed information on the datasets used is shown in Table 1.

Table 1. Datasets used for BCtypeFinder evaluation.

	Dataset	# of CpGs	# of samples	with subtype label	Use label for training
Source	TCGA-BRCA	27K	267	✓	✓
		450K	793	✓	✓
Target	GSE69914	450K	305	X	X
	GSE75067	450K	188	X	X
	GSE72245	450K	118	✓	X

Data preprocessing followed a similar approach to meth-SemiCancer [11]. Initially, genes shared between the source and target datasets were extracted. CpG sites with more than 20% missing values were eliminated to mitigate bias during model training. Median imputation was performed for the remaining missing values, and the top 2,000 highly varying CpGs for the source dataset were selected.

3.2 Hyperparameter Setting

In BCtypeFinder, all modules comprise two fully connected layers, with the feature extractor having 1024 and 512 hidden nodes, and both the domain discriminator and subtype classifier having 256 and 64 hidden nodes, followed by a softmax layer. The adaptive optimization algorithm Adam [19] was employed for training BCtypeFinder, with learning rates set at 10^{-4}, 10^{-5}, and 10^{-6} for the feature extractor, subtype classifier, and domain discriminator, respectively. The training process consisted of 500 epochs for both pre-training and adversarial training phases. During fine-tuning, BCtypeFinder underwent an initial training of 500 epochs based on the SSL loss to establish stable pseudo-labels for the target dataset. Subsequently, it was iteratively optimized for 800 epochs based on the SSL and class alignment loss. In the SSL loss, α_f was set to 0.01, and pseudo-labels were updated in each iteration. BCtypeFinder was implemented using the PyTorch library (Version 1.6.0).

4 Results

4.1 Performance Evaluation of BCtypeFinder

BCtypeFinder was compared with meth-SemiCancer, a recently proposed DNA methylation-based cancer subtype prediction model, and widely-used machine learning-based classifiers: Support Vector Machine (SVM), Random Forest (RF), and Logistic Regression (LR). The comparison methods were optimized on the TCGA-BRCA dataset, with training and testing datasets randomly split in an 8:2 ratio. Each experiment was repeated five times, and hyperparameters yielding the highest average accuracy for the testing dataset were selected (Supplementary Table S1[1]). Optimized hyperparameter settings for each classifier are as follows: meth-SemiCancer (Two layers of hidden nodes = 1000-500, learning rate = 1e−3, alpha = 0.05, training epochs = 3000), SVM (kernel = linear, C = 2^{-3}), RF (criterion = gini, estimators = 100, min_samples_leaf = 3), and LR (C = 2^{-1}, max_iter = 100). Evaluation metrics included accuracy, weighted F1-score, Matthews Correlation Coefficient (MCC), and Area under the ROC Curve (AUC).

The first evaluation utilized the TCGA-BRCA dataset as the source data and the three GEO datasets as the unlabeled target data. Testing was performed on the 87 samples in the target dataset with subtype labels (GSE72245). The experiment was repeated five times, and BCtypeFinder consistently outperformed the other methods, achieving an average accuracy of 0.816 (Fig. 2a, Supplementary Table S2). Meth-SemiCancer exhibited the second-best performance with an average accuracy of 0.736, while SVM, RF, and LR had average accuracies of 0.655, 0.545, and 0.667, respectively. BCtypeFinder also yielded the best average F1-score, MCC, and AUC, reaching 0.822, 0.752, and 0.883, respectively, outperforming meth-SemiCancer with the second-highest average performance of 0.748, 0.664, and 0.848. Runtime comparison showed that BCtypeFinder has lower average running time of 580 s, as compared to the averaged 877 s of meth-SemiCancer, underscoring the enhanced computational efficiency of our proposed method.

Next, 10-fold cross-validation was performed on the TCGA-BRCA dataset, with the three GEO datasets (GSE69914, GSE75067, and GSE72245) as the target data (Fig. 2b, Supplementary Table S3). BCtypeFinder demonstrated improved breast cancer subtype classification, achieving the highest average accuracy and F1-score of 0.849 and

[1] Supplementary data are available at https://github.com/joungmin-choi/BCtypeFinder.

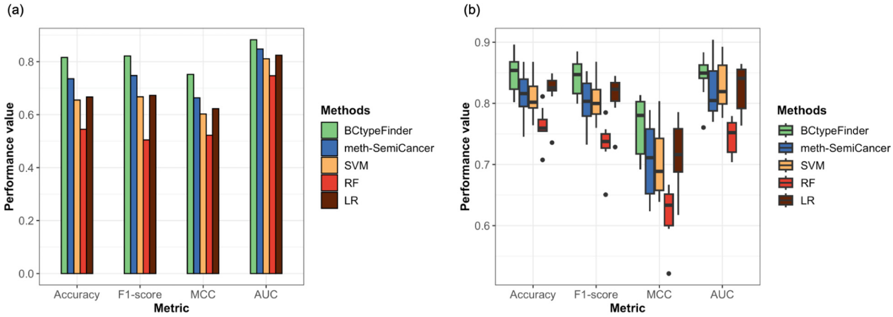


Fig. 2. Performance comparison of BCtypeFinder with other methods in Breast cancer subtype prediction. (a) Average classification performance results for 87 samples of the GSE72245 dataset. (b) 10-fold cross validation results on the TCGA-BRCA dataset.

0.843, respectively, compared to the second-best values of 0.821 and 0.814. BCtypeFinder exhibited stable classification with less variation in performance across 10-fold cross-validation compared to SSL-based meth-SemiCancer. These results underscore the robustness of BCtypeFinder for breast cancer subtype prediction in DNA methylation datasets, achieved through subtype alignment and discrimination across batches.

4.2 Exploring Batch Effect Correction in BCtypeFinder

BCtypeFinder employs adversarial training and subtype alignment to mitigate batch effects in various methylation datasets, facilitating the discrimination of subtypes. To visually evaluate the impact of these training methods, we applied the uniform manifold approximation and projection (UMAP) technique. The features extracted by BCtypeFinder were compressed into two-dimensional spaces and annotated for each respective dataset. Additionally, UMAP visualization was performed on the TCGA-BRCA and GSE72245 datasets that have subtype labels. Figure 3 illustrates the UMAP visualization of BCtypeFinder-extracted features alongside the uncorrected dataset, representing the preprocessed original dataset.

Visualization of the uncorrected data reveals significant divergence among breast cancer patients across samples, forming distinct groups based on batch, while patients with different subtypes appear mixed and scattered. After batch effect smoothing by BCtypeFinder, distinctions based on batches were eliminated, revealing a clear separation of patients with the same subtype across different batches. Patients within the same breast cancer subtype were distinctly organized into individual clusters. These results underscore BCtypeFinder's efficacy in effectively mitigating batch effects and aligning the distributions of identical breast cancer subtypes across various datasets.

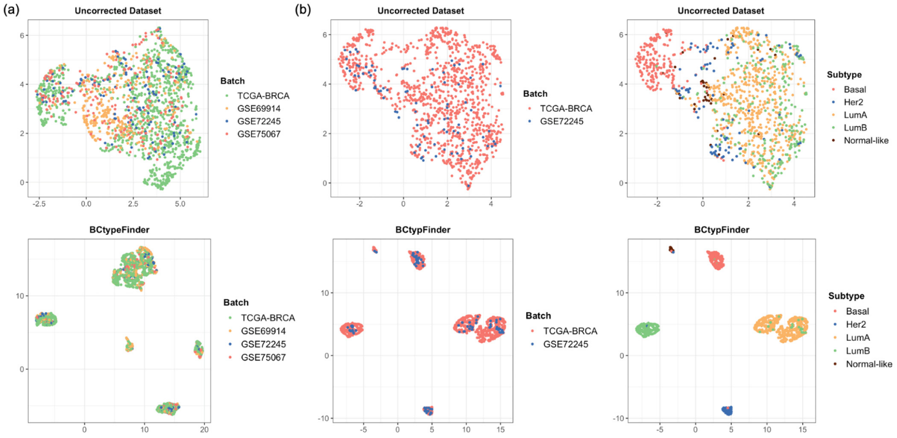


Fig. 3. UMAP visualization for the source and target datasets comparing the uncorrected dataset and the features extracted from BCtypeFinder. (a) UMAP plots colored by batch for the TCGA-BRCA source dataset and three GEO datasets used as target. (b) UMAP plots colored by batch and the breast cancer subtypes for the source dataset and 87 samples of GSE72245 target dataset having subtype labels.

4.3 Effectiveness of Each Module in BCtypeFinder

To assess the performance improvements achieved by each training phase and understand how predictions evolve throughout the training process, we measured the testing classification performance after each phase. For this experiment, BCtypeFinder was trained using TCGA-BRCA as the source dataset and three GEO datasets as the target datasets. Testing was performed on the GSE72245 dataset that contains 87 labeled samples. The results, summarized in Supplementary Table S4, indicate that optimizing the proposed model through pre-training and the SSL loss led to a slight improvement in classification performance (Average accuracy of 0.720 for pre-training to 0.730 for SSL). However, this improvement was not significant, even when the target dataset was utilized during SSL. Notably, when the model underwent iterative fine-tuning with SSL and subtype alignment, a substantial performance increase was observed across all metrics. Specifically, accuracy improved from 0.730 to 0.816, and the F1-score increased from 0.723 to 0.822. These findings highlight the effectiveness of subtype alignment training in harmonizing identical breast cancer subtypes across batches. By generating more accurate pseudo labels, this approach enhances the classifier's discriminative power, thereby playing a crucial role in the accurate prediction of breast cancer subtypes.

5 Discussion and Conclusions

In this study, we introduced BCtypeFinder, a breast cancer subtype prediction model leveraging domain adaptation with semi-supervised learning and DNA methylation profiles. The model underwent a multi-phase training process: pre-training on the source dataset for weight initialization, adversarial training to extract domain-invariant features, and fine-tuning involving SSL for pseudo-label generation and subtype distribution alignment across datasets. BCtypeFinder was evaluated against meth-SemiCancer, a state-of-the-art DNA methylome-based cancer subtype classifier, as well as commonly used ML-based classifiers. Our model exhibited superior performance with the highest average accuracy and AUC, demonstrating robust classification capabilities. Furthermore, we explored the impact of adversarial training and subtype alignment on batch correction in different DNA methylation datasets. BCtypeFinder successfully eliminated batch distinctions among patients with the same subtype across batches, enhancing the classifier's discriminative ability for breast cancer subtypes. Overall, these findings demonstrate that BCtypeFinder can serve as the initial assessment of breast cancer subtypes, thus facilitating downstream clinical diagnosis and personalized treatment.

While BCtypeFinder demonstrated performance improvements, there remain certain limitations that warrant further enhancement. Computational resource constraints led to feature selection to reduce the training feature set. Future experiments should explore alternative feature selection approaches. Additionally, we aim to extend BCtypeFinder's applicability to other cancers through extensive testing and optimization experiments in the future.

Acknowledgments. This work is partially supported by the ICTAS EFO-O Investment Seed Grant Program at Virginia Tech.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Sung, H., Ferlay, J., Siegel, R.L., et al.: Global cancer statistics 2020: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries. CA Cancer J. Clin. **71**(3), 209–249 (2021)
2. Guo, L., Kong, D., Liu, J., et al.: Breast cancer heterogeneity and its implication in personalized precision therapy. Exp. Hematol. Oncol. **12**(1), 1–27 (2023)
3. Fisher, R., Pusztai, L., Swanton, C.: Cancer heterogeneity: implications for targeted therapeutics. Br. J. Cancer **108**(3), 479–485 (2013)
4. Parker, J.S., Mullins, M., Cheang, M.C., et al.: Supervised risk predictor of breast cancer based on intrinsic subtypes. J. Clin. Oncol. **27**(8), 1160 (2009)
5. Pu, M., Messer, K., Davies, S.R., et al.: Based PAM50 signature and long-term breast cancer survival. Breast Cancer Res. Treat. **179**, 197–206 (2020)
6. Titus, A.J., Way, G.P., Johnson, K.C., Christensen, B.C.: Deconvolution of DNA methylation identifies differentially methylated gene regions on 1p36 across breast cancer subtypes. Sci. Rep. **7**(1), 11594 (2017)
7. Ehrlich, M., Lacey, M.: Epigenetic alterations in oncogenesis. In: Epigenetic Alterations in Oncogenesis, Advances in Experimental, vol. 31, pp. 31–56 (2013)
8. Lakshminarasimhan, R., Liang, G.: The role of DNA methylation in cancer. In: Jeltsch, A., Jurkowska, R. (eds.) DNA Methyltransferases - Role and Function. AEMB, vol. 945, pp. 151–172. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-43624-1_7

9. Kim, A., Mo, K., Kwon, H., et al.: Epigenetic regulation in breast cancer: insights on epidrugs. Epigenomes **7**(1), 6 (2023)
10. Zhang, S., Wang, Y., Gu, Y., et al.: Specific breast cancer prognosis-subtype distinctions based on DNA methylation patterns. Mol. Oncol. **12**(7), 1047–1060 (2018)
11. Choi, J.M., Park, C., Chae, H.: Meth-semicancer: a cancer subtype classification framework via semi-supervised learning utilizing DNA methylation profiles. BMC Bioinform. **24**(1), 1–14 (2023)
12. Zhang, Y.: A survey of unsupervised domain adaptation for visual recognition. arXiv preprint arXiv:2112.06745 (2021)
13. Zhou, X., Chai, H., Zeng, Y., Zhao, H., Yang, Y.: scAdapt: virtual adversarial domain adaptation network for single cell RNA-seq data classification across platforms and species. Briefings Bioinform. **22**(6), bbab281 (2021)
14. Bae, S., Na, K.J., Koh, J., et al.: CellDART: cell type inference by domain adaptation of single-cell and spatial transcriptomic data. Nucleic Acids Res. **50**(10), e57–e57 (2022)
15. Tomczak, K., Czerwińska, P., Wiznerowicz, M.: Review the cancer genome atlas (TCGA): an immeasurable source of knowledge. Contemp. Oncol. Współczesna Onkologia **2015**(1), 68–77 (2015)
16. Teschendorff, A.E., Gao, Y., Jones, A., et al.: DNA methylation outliers in normal breast tissue identify field defects that are enriched in cancer. Nat. Commun. **7**(1), 10478 (2016)
17. Holm, K., Staaf, J., Lauss, M., et al.: An integrated genomics analysis of epigenetic subtypes in human breast tumors links DNA methylation patterns to chromatin states in normal mammary cells. Breast Cancer Res. **18**, 1–20 (2016)
18. Jeschke, J., Bizet, M., Desmedt, C., et al.: DNA methylation–based immune response signature improves patient diagnosis in multiple cancers. J. Clin. Investig. **127**(8), 3090–3102 (2017)
19. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)

CI-VAE for Single-Cell: Leveraging Generative-AI to Enhance Disease Understanding

Mohsen Nabian(✉), Zahra Eftekhari, and Chi Wah Wong

City of Hope, San Jose, USA
mnabian@coh.org

Abstract. Understanding cellular disease processes like cancer is key for improving diagnosis and treatment. Single-cell RNA sequencing (scRNA-seq) enables modeling transitions between normal and diseased cellular states in complex tissues. However, interpolating between healthy and diseased states in high-dimensional scRNA-seq data poses computational challenges. We use the Class-Informed Variational Autoencoder (CI-VAE), a generative AI model, to learn low-dimensional cell-type-specific representations from scRNA-seq data. During inference, CI-VAE interpolates between normal and diseased cells, robustly predicting cell-type-specific gene expression trajectories from healthy to disease states. Applied to colon cancer data, CI-VAE closely predicted observed transitions by generating synthetic gene expression changes associated with cancer progression for each cell type, potentially offering insights into underlying molecular mechanisms for disease understanding, biomarker discovery, and targeted therapy design.

Keywords: Generative AI · Single Cell RNAseq · VAE · CI-VAE · Deep Learning · Disease Understanding

1 Introduction

Single-cell RNA sequencing (scRNA-seq) has emerged as a powerful tool for unveiling the cellular diversity within complex tissues, including cancer [9, 10]. By capturing the transcriptional profiles of individual cells, scRNA-seq offers unprecedented insights into the molecular mechanisms underlying disease progression. However, the inherent high-dimensionality and sparsity of scRNA-seq data pose significant challenges for computational analysis, particularly in modeling the subtle transitions between healthy and diseased cellular states [2, 3, 11].

Variational Autoencoders (VAEs) [1, 4, 5, 12] have shown promise in efficiently encoding high-dimensional data into a lower-dimensional latent space which allows for generating new data within the same distribution as the original data.

In this work, we used the Class-Informed Variational Autoencoder (CI-VAE) [7, 8], a variant of VAEs that incorporate class information of data (, cell-types in single-cell data)

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 367–372, 2024.
https://doi.org/10.1007/978-3-031-66538-7_36

into constructing the generative latent space. Unlike traditional VAEs, CI-VAE employs a linear discriminator that operates on the latent representations, enhancing the model's ability to generate cell-type-specific synthetic cells during interpolation, particularly in the context of studying cellular transitions from normal to diseased conditions.

2 Methodology

2.1 Autoencoders and Variational Autoencoders (VAEs)

Autoencoders are a class of unsupervised neural networks consisting of an encoder and a decoder. The encoder maps input data x into a lower-dimensional latent space z, while the decoder attempts to reconstruct the input data from this latent space, producing $\hat{x}$. Variational Autoencoders (VAEs) enhance this architecture by mapping inputs to a posterior distribution $p(z|x)$, instead of a direct encoding to z. The total cost function for VAEs includes a reconstruction error and a regularization term that encourages the latent space distribution to approximate a standard Gaussian.

2.2 Class-Informed Variational Autoencoders (CI-VAE)

The CI-VAE model [7, 8] introduces a supervised dimension to the VAE framework by incorporating a linear discriminator within its architecture. This discriminator ensures observations from different classes remain separable within the latent space. The CI-VAE's cost function includes the cross-entropy loss from the discriminator, in addition to the reconstruction error and the KL divergence term.

2.3 Application to Single-Cell RNA Sequencing (ScRNA-Seq) Data

In the context of CI-VAE, each cell is treated as an observation, with the cell type serving as the class of the data. During training, CI-VAE learns the underlying low-dimensional latent space of all cells with the additional objective of forming the latent space to be linearly separable across different cell types. During inference, the objective shifts to interpolating between two cells within the same cell type, one healthy and the other diseased, to elucidate the mechanisms underlying disease development.

3 Results and Discussion

3.1 CI-VAE for Understanding Colon Cancer

We demonstrate the application of CI-VAE to single-cell RNA-seq data of colon cancer [6]. This dataset consists of 20K genes for 25K individual cells from colon cancer tissue, with 22 identified cell types. For each cell type, normal, borderline tumor, and core tumor cells are present.

We used CI-VAE to interpolate from normal cells to cancer cells across different cell types, generating synthetic data that captures the gene expression trajectories from healthy to diseased states. To validate the model's predictions, we compared the generated trajectories to the ground truth data, specifically the borderline tumor cells (Fig. 1).

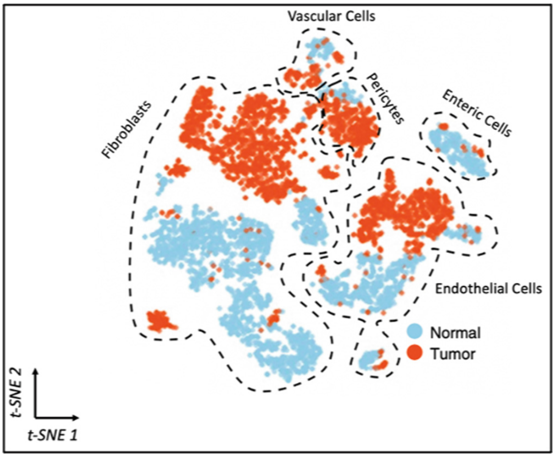


Fig. 1. 2D TSNE plot of high dimensional RNAseq data of cells in colon cancer tissue. Each dot represents one cell. Cells are grouped by cell types and blue color represent normal cells and orange color represent core tumor cells. The figure is extracted from [6].

As shown in Figures 2, 3 and 4, the gene expression trajectories generated by CI-VAE closely match the observed transitions from normal to borderline tumor to core tumor cells, across multiple cell types and genes. This correspondence indicates that CI-VAE can potentially model the molecular mechanisms underlying cancer development, potentially informing the identification of early biomarkers and the design of targeted therapies.

4 Discussion

The results of this study demonstrate the utility of the Class-Informed Variational Autoencoder (CI-VAE) model in analyzing single-cell RNA sequencing (scRNA-seq) data to elucidate the molecular trajectories underlying the transition from normal to cancerous cell states. By incorporating class information into the VAE framework, CI-VAE was able to effectively capture the subtle variations between cell types and states, enabling the generation of synthetic data that closely approximates the observed gene expression changes.

The application of CI-VAE to colon cancer scRNA-seq data provided valuable insights into the progression of the disease at the cellular level. By interpolating between normal and cancerous cells within specific cell types, CI-VAE generated trajectories that aligned well with the observed gene expression patterns, in particular, the intermediate "borderline" tumor cell states. This ability to model the gradual transition from healthy to diseased conditions offers an opportunity to gain a more nuanced understanding of the molecular mechanisms driving cancer development.

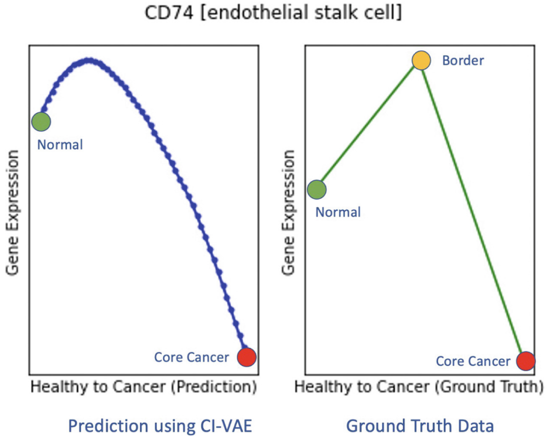


Fig. 2. Left, predicted RNA expression from normal cell to core tumor cancer cell using CI-VAE, Right, Ground Truth trajectory with Normal, Borderline tumor and Core Cancer tumor. Results are projected for endothelial cell type and for CD74 Gene. These results are the median of 100 random traversals from normal cells to tumor cells. Our predictions shows that CD74 is over expressed in the onset of cancer development and then under-expressed over the development of core cancer cells.

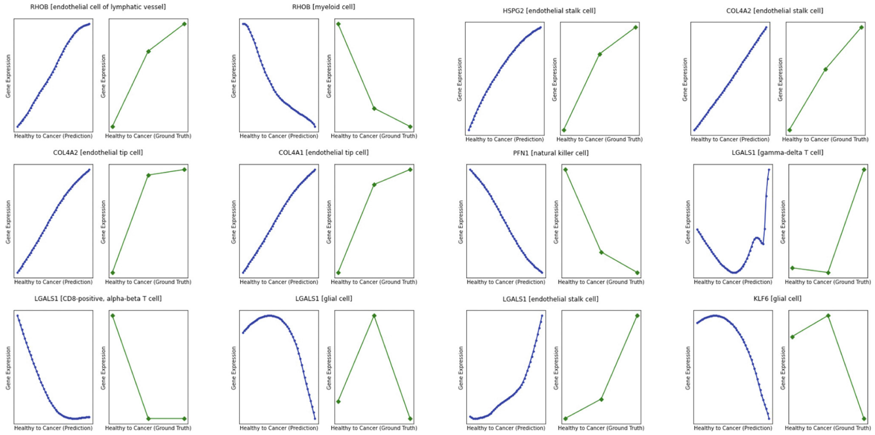


Fig. 3. Left, predicted RNA expression from normal cell to tumor cell using CI-VAE, Right, Ground Truth. Each plot pair is for a specific cell type and a specific gene.

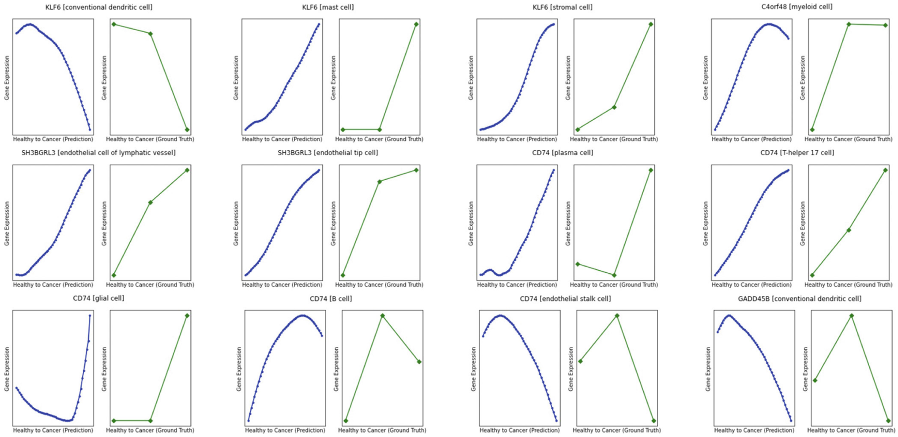


Fig. 4. Left, predicted RNA expression from normal cell to tumor cell using CI-VAE, Right, Ground Truth. Each plot pair is for a specific cell type and a specific gene.

The correspondence between the CI-VAE-generated trajectories and the ground truth data underscores the potential of this approach to enhance our knowledge of disease progression. By capturing the dynamic changes in gene expression across the spectrum from normal to cancerous states, CI-VAE may potentially inform the identification of early biomarkers and therapeutic targets, as well as enable the design of more effective interventions tailored to specific cell types and stages of the disease.

5 Conclusion

This work demonstrates the potential of using generative AI and in particular, the CI-VAE model to advance our understanding of the molecular underpinnings of cancer development through the analysis of single-cell RNA sequencing data. By providing a computational framework that can effectively capture the dynamics of cellular states, CI-VAE potentially provides a valuable tool for uncovering novel insights into complex biological processes in disease development and informing the development of targeted therapeutic strategies.

References

1. Dilokthanakul, N., et al.: Deep unsupervised clustering with Gaussian mixture variational autoencoders. arXiv preprint arXiv:1611.02648 (2016)
2. Ding, J., Regev, A.: Deep generative model embedding of single-cell RNA-seq profiles on hyperspheres and hyperbolic spaces. Nat. Commun. **12**(1), 2554 (2021)
3. Hie, B., Peters, J., Nyquist, S.K., Shalek, A.K., Berger, B., Bryson, B.D.: Computational methods for single-cell RNA sequencing. Ann. Rev. Biomed. Data Sci. **3**, 339–364 (2020)
4. Hsu, W.N., Zhang, Y., Glass, J.: Unsupervised domain adaptation for robust speech recognition via variational autoencoder-based data augmentation. In: 2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), pp. 16–23. IEEE (2017)

5. Kingma, D.P., Welling, M.: Auto-encoding variational Bayes. arXiv preprint arXiv:1312.6114 (2013)
6. Lee, H.O., et al.: Lineage-dependent gene expression programs influence the immune landscape of colorectal cancer. Nat. Genet. **52**(6), 594–603 (2020)
7. Nabian, M., Eftekhari, Z., Wong, A.: CI-VAE: a class-informed deep variational autoencoder for enhanced class-specific data interpolation (2022)
8. Nabian, M., Eftekhari, Z., Wong, C.W.: CI-VAE: a generative deep learning model for class-specific data interpolation (2023)
9. Patel, A.P., et al.: Single-cell RNA-seq highlights intratumoral heterogeneity in primary glioblastoma. Science **344**(6190), 1396–1401 (2014)
10. Saliba, A.E., Westermann, A.J., Gorski, S.A., Vogel, J.: Single-cell RNA-seq: advances and future challenges. Nucleic Acids Res. **42**(14), 8845–8860 (2014)
11. Sun, S., Zhu, J., Ma, Y., Zhou, X.: Accuracy, robustness and scalability of dimensionality reduction methods for single-cell RNA-seq analysis. Genome Biol. **20**, 1–21 (2019)
12. Svensson, V., Gayoso, A., Yosef, N., Pachter, L.: Interpretable factor models of single-cell RNA-seq via variational autoencoders. Bioinformatics **36**(11), 3418–3421 (2020)

ProteinEngine: Empower LLM with Domain Knowledge for Protein Engineering

Yiqing Shen[1], Outongyi Lv[2], Houying Zhu[3], and Yu Guang Wang[2](✉)

[1] Department of Computer Science, Johns Hopkins University, Baltimore, USA
yshen92@jhu.edu

[2] Institute of Natural Sciences, Shanghai Jiao Tong University, Shanghai, China
yuguang.wang@sjtu.edu.cn

[3] School of Mathematical and Physical Sciences, Macquarie University, Sydney, Australia

Abstract. Large language models (LLMs) have garnered considerable attention for their proficiency in tackling intricate tasks, particularly leveraging their capacities for zero-shot and in-context learning. However, their utility has been predominantly restricted to general tasks due to an absence of domain-specific knowledge. This constraint becomes particularly pertinent in the realm of protein engineering, where specialized expertise is required for tasks such as protein function prediction, protein evolution analysis, and protein design, with a level of specialization that existing LLMs cannot furnish. In response to this challenge, we introduce ProteinEngine, a human-centered platform aimed at amplifying the capabilities of LLMs in protein engineering by seamlessly integrating a comprehensive range of relevant tools, packages, and software via API calls. Uniquely, ProteinEngine assigns three distinct roles to LLMs, facilitating efficient task delegation, specialized task resolution, and effective communication of results. This design fosters high extensibility and promotes the smooth incorporation of new algorithms, models, and features for future development. Extensive user studies, involving participants from both the AI and protein engineering communities across academia and industry, consistently validate the superiority of ProteinEngine in augmenting the reliability and precision of deep learning in protein engineering tasks. Consequently, our findings highlight the potential of ProteinEngine to bride the disconnected tools for future research in the protein engineering domain.

Keyword: Deep Learning · Large Language Model · Protein Design · AI for Protein Design

1 Introduction

Large language models (LLMs) have achieved remarkable successes in solving complex tasks, showcasing their zero-shot learning capabilities [18]. However, the effectiveness of these models tends to plateau when faced with more specialized tasks due to their inability to access domain-specific knowledge or utilize specialized tools tailored for certain applications. This limitation becomes glaringly apparent in the context of protein engineering tasks. Although LLMs have been explored for specific tasks within the

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 373–383, 2024.
https://doi.org/10.1007/978-3-031-66538-7_37

protein engineering domain such as protein structure prediction [11], protein evolution analysis, or de novo protein design [6, 13], their application often demands significant alterations to the model architecture. These modifications, coupled with the need for learning from scratch using domain-specific datasets, present two-fold challenges. First, this approach underutilizes the capabilities of well-trained foundation models, given their robust pre-existing knowledge base. Second, the process of specialization often leads to the loss of the model's conversational abilities, a key feature that makes LLMs versatile and user-friendly.

The introduction of in-context learning capabilities in LLMs [4] has ushered potential solutions to the enduring challenge of domain knowledge scarcity [4]. In this new paradigm, the LLM acts as a centralized, cognitive-like system, which is capable of addressing domain-specific tasks by invoking relevant Application Programming Interfaces (APIs) or systems to bride the knowledge gap within the specific domain. However, despite these advancements, most existing solutions tend to restrict the tools they incorporate to commonly used APIs such as calculators, calendars, and web searches, or relatively simple AI models like text-to-image generation models. In the context of protein engineering, both the task formulation and the APIs involved, as well as the AI models, manifest greater complexity. They are characterized by a diverse modality of input and a larger set of arguments, accentuating the need for more sophisticated and flexible systems. Consequently, further research and development are required to fully leverage the potential of LLMs in complex domains such as protein engineering. To narrow the gap, we present ProteinEngine, a novel LLM system for protein engineering. The major contributions are three-fold:

(1) We introduce ProteinEngine, a human-centered platform to augment the capabilities of LLMs in tackling protein engineering tasks. This is achieved by seamlessly integrating a comprehensive array of tools, packages, and software relevant to protein engineering, all accessible through APIs.
(2) We propose a role-playing framework, comprising AI Project Manager (AI-PM), AI Domain Expert (AI-DE), and AI Presenter (AI-Pr) modules, which facilitates eficient task delegation, promotes interdisciplinary integration, ensures dynamic adaptability, and enables effective communication of results respectively. This design principle provides substantial flexibility, allowing for easy integration and extensibility with emerging AI protein-design models.
(3) Through comprehensive user studies, we demonstrate the superior performance of ProteinEngine in not only enhancing the usability of currently disconnected protein engineering tools but also reducing the workload and learning difficulty across users with different backgrounds.

2 ProteinEngine: LLM for Protein Engineering

Method Overview Our proposed ProteinEngine is a human-centered system designed to augment the capabilities of existing LLMs to address a broader spectrum of intricate protein engineering tasks. Specifically, we assign three critical roles to the LLMs: the AI Project Manager (AI-PM), the AI Domain Expert (AI-DE), and the AI Presenter (AI-Pr). The overall pipeline illustrating these roles and their interactions is depicted in

Fig. 1. The AI-PM functions as the primary coordinator, interpreting user input expressed in natural language and ensuring all necessary inputs, configurations, arguments, and conditions are correctly provided. Subsequently, it breaks down a given complex task into smaller, more manageable sub-tasks, delegating them to the appropriate AI-DEs. Then, we employ multiple AI-DEs within the platform to address the wide variety of challenges inherent in protein engineering and to facilitate future expansion. Each AI-DE specializes in a particular domain or category of tasks, ensuring a comprehensive coverage of the diverse aspects of protein engineering. During the inference stage, the AI-PM selects and assigns a subset of all AI-DEs to execute relevant APIs based on the nature of the task at hand. Lastly, the AI-Pr is tasked with presenting the results, either unimodal or multimodal, generated by the AI-DEs to the user.

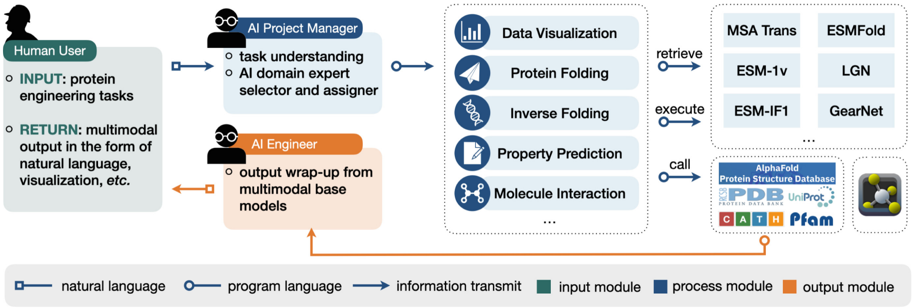


Fig. 1. The overall framework of the proposed ProteinEngine, which incorporates three distinct roles, each assigned to a separate LLM.

AI Project Manager. The LLM performing as the AI-PM acts as the primary interface, bridging the gap between the user and the underlying protein engineering tools within the ProteinEngine platform. In its core role, the AI-PM is tasked with interpreting user input presented in natural language, discerning the context, and identifying the necessary tasks to be performed. Beyond this, the AI-PM ensures that all required inputs, arguments, configurations, and conditions are correctly provided. To accurately parse and deconstruct the user's query, the AI-PM uses in-context learning, a more efficient alternative to the computationally demanding process of LLM fine-tuning. By systematically decomposing complex tasks into smaller, manageable sub-tasks, the AI-PM ensures a thorough understanding of the user's requirements. Once the sub-tasks are defined, the AI-PM delegates them to the appropriate AI-DEs, taking into account their respective areas of specialization.

AI Domain Expert. The AI-DE in the ProteinEngine is specifically designed to manage a distinct category of tasks pertaining to protein engineering. A team of multiple AI-DEs is assembled, with each expert equipped with the necessary domain-specific knowledge and tools to execute its designated tasks. To ensure the AI-DEs perform efficiently and adaptively, we have implemented a novel self-feedback communication loop mechanism between the AI-DE and the AI-PM. This autonomous mechanism, which operates without the need for human intervention, enables AI-DEs to progressively refine their understanding of new challenges that may arise during the execution process, and

Table 1. The involved AI models and APIs for protein engineering in the proposed ProteinEngine.

API	Functionality	Description	Input	Output
AlphaFold 2 [9]	protein folding	single-chain structure prediction with MSA	protein sequence	atom-level 3D coordinates; residue-level pLDDT
AlphaFold-Multimer [5]	protein folding	multi-chain structure prediction with MSA	protein sequence	atom-level 3D coordinates; residue-level pLDDT
ESMFold [11]	protein folding	MLM-based structure prediction without MSA	protein sequence	atom-level 3D coordinates; residue-level pLDDT
MSA Transformer [14]	protein folding	single-chain structure prediction with MSA	multiple sequence alignment	atom-level 3D coordinates; in .pdb format
ESM-IF1 [11]	inverse folding	single-site mutation Transformer-based	protein sequence	de novo protein sequence
LGN [19]	variant effect prediction	deep mutation GNN based denoising	protein graph	de novo protein sequence
Equidock [7]	protein-target interaction	rigid-body docking	two protein structures in .pdb format	binding affinity score
EquiBind [15]	protein-target interaction	rigid-body docking	protein-ligand structure in .pdb format	protein-ligand interaction sites binding affinity score
DiffDock [3]	protein-target interaction	rigid-body docking	antibody-antigen structures in .pdb format	bound structure of complex
Diffab [12]	protein target interaction	antibody-antigen interaction	antibody-antibody structures in .pdb format	bound structure of complex binding affinity, epitope mapping

(*continued*)

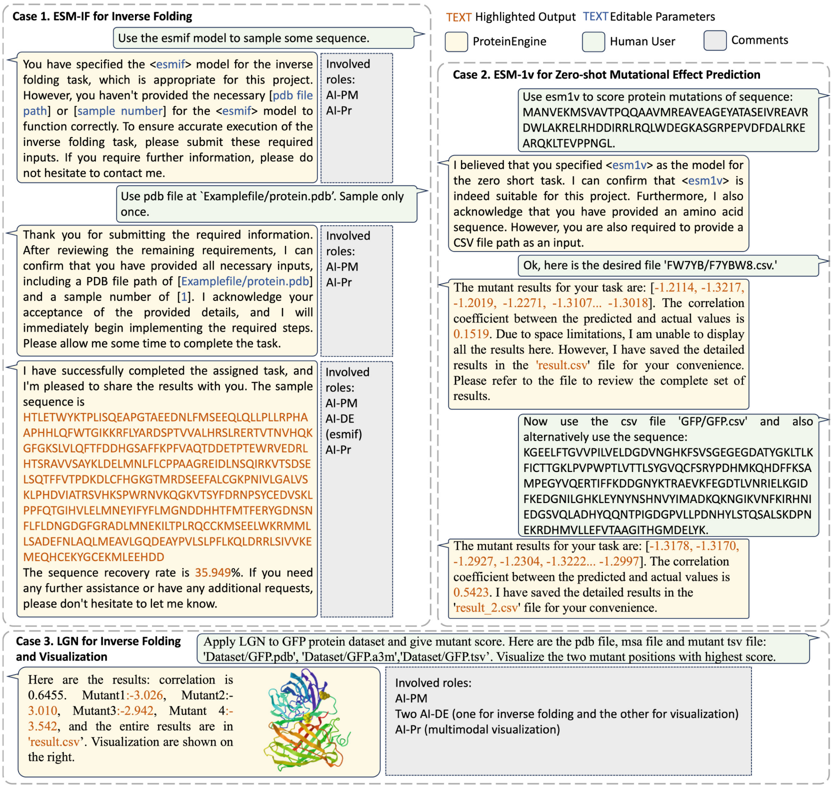
Table 1. (*continued*)

API	Functionality	Description	Input	Output
ProtENN [1]	sequence generation	language-based model	protein sequence and structure	de novo protein sequence with function
Progen [13]	sequence generation	language-based model	protein sequence	de novo protein sequence with function
Grade-IF [16]	sequence generation	graph-based model	protein sequence	de novo protein sequence
GearNet [17]	property prediction	latent representation of protein structure	protein graph	function or structure label
DeepSol [10]	property prediction	solubility prediction	protein sequence	solubility
PyMOL	protein visualization	visualize 3D conformation for a given protein molecule	.pdb document	3D visualization of the protein
VMD	protein visualization	visualize 3D conformation for a given protein molecule	.pdb document	3D visualization of the protein
BioMedLM	biomedical domain Q&A	trained on biomedical literature and clinical notes	natural language on biomedicine or healthcare	answer questions as a specialist in the field

to seek assistance from their fellow AI-DEs, if required. As a result, AI-DEs can dynamically adjust and respond to the evolving demands of the tasks, thereby maintaining a high degree of accuracy and effectiveness.

AI Presenter. AI-Pr aggregates and presents the results generated by the AI-DEs in a clear, concise, and user-friendly manner, ensuring the user can easily interpret and utilize the generated insights, fostering a deeper understanding. To effectively communicate the results to the user, the AI-Pr is capable of visualizing multimodal data, which includes, but is not limited to, images and textual data. This presentation is tailored to cater to different user preferences, and it enhances the comprehensibility of the data, enabling users to quickly grasp the key insights and outcomes delivered by the AI-DEs.

APIs for Protein Engineering. The APIs integrated within our system, along with their corresponding task category taxonomy, are delineated in Table 1, which covers



TEXT Highlighted Output TEXT Editable Parameters

□ ProteinEngine □ Human User □ Comments

Case 1. ESM-IF for Inverse Folding

Use the esmif model to sample some sequence.

You have specified the <esmif> model for the inverse folding task, which is appropriate for this project. However, you haven't provided the necessary [pdb file path] or [sample number] for the <esmif> model to function correctly. To ensure accurate execution of the inverse folding task, please submit these required inputs. If you require further information, please do not hesitate to contact me.

Involved roles:
AI-PM
AI-Pr

Use pdb file at `Examplefile/protein.pdb’. Sample only once.

Thank you for submitting the required information. After reviewing the remaining requirements, I can confirm that you have provided all necessary inputs, including a PDB file path of [Examplefile/protein.pdb] and a sample number of [1]. I acknowledge your acceptance of the provided details, and I will immediately begin implementing the required steps. Please allow me some time to complete the task.

Involved roles:
AI-PM
AI-Pr

I have successfully completed the assigned task, and I'm pleased to share the results with you. The sample sequence is
HTLETWYKTPLISQEAPGTAEEDNLFMSEEQLQLLPLLRPHA
APHHLQFWTGIKKRFLYARDSPTVVALHRSLRERTVTNVHQK
GFGKSLVLQFTFDDHGSAFFKPFVAQTDDETPTEWRVEDRL
HTSRAVVSAYKLDELMNLFLCPPAAGREIDLNSQIRKVTSDSE
LSQTFFVTPDKDLCFHGKGTMRDSEEFALCGKPNIVLGALVS
KLPHDVIATRSVHKSPWRNVKQGKVTSYFDRNPSYCEDVSKL
PPFQTGIHVLELMNEYIFYFLMGNDDHHTFMTFERYGDNSN
FLFLDNGDGFGRADLMNEKILTPLRQCCKMSEELWKRMML
LSADEFNLAQLMEAVLGQDEAYPVLSLPFLKQLDRRLSIVVKE
MEQHCEKYGCEKMLEEHDD
The sequence recovery rate is 35.949%. If you need any further assistance or have any additional requests, please don't hesitate to let me know.

Involved roles:
AI-PM
AI-DE
(esmif)
AI-Pr

Case 2. ESM-1v for Zero-shot Mutational Effect Prediction

Use esm1v to score protein mutations of sequence:
MANVEKMSVAVTPQQAAVMREAVEAGEYATASEIVREAVR
DWLAKRELRHDDIRRLRQLWDEGKASGRPEPVDFDALRKE
ARQKLTEVPPNGL.

I believed that you specified <esm1v> as the model for the zero short task. I can confirm that <esm1v> is indeed suitable for this project. Furthermore, I also acknowledge that you have provided an amino acid sequence. However, you are also required to provide a CSV file path as an input.

Ok, here is the desired file 'FW7YB/F7YBW8.csv.'

The mutant results for your task are: [-1.2114, -1.3217, -1.2019, -1.2271, -1.3107... -1.3018]. The correlation coefficient between the predicted and actual values is 0.1519. Due to space limitations, I am unable to display all the results here. However, I have saved the detailed results in the 'result.csv' file for your convenience. Please refer to the file to review the complete set of results.

Now use the csv file 'GFP/GFP.csv' and also alternatively use the sequence:
KGEELFTGVVPILVELDGDVNGHKFSVSGEGEGDATYGKLTLK
FICTTGKLPVPWPTLVTTLSYGVQCFSRYPDHMKQHDFFKSA
MPEGYVQERTIFFKDDGNYKTRAEVKFEGDTLVNRIELKGID
FKEDGNILGHKLEYNYNSHNVYIMADKQKNGIKVNFKIRHNI
EDGSVQLADHYQQNTPIGDGPVLLPDNHYLSTQSALSKDPN
EKRDHMVLLEFVTAAGITHGMDELYK.

The mutant results for your task are: [-1.3178, -1.3170, -1.2927, -1.2304, -1.3222... -1.2997]. The correlation coefficient between the predicted and actual values is 0.5423. I have saved the detailed results in the 'result_2.csv' file for your convenience.

Case 3. LGN for Inverse Folding and Visualization

Apply LGN to GFP protein dataset and give mutant score. Here are the pdb file, msa file and mutant tsv file: 'Dataset/GFP.pdb', 'Dataset/GFP.a3m','Dataset/GFP.tsv’. Visualize the two mutant positions with highest score.

Here are the results: correlation is 0.6455. Mutant1:-3.026, Mutant2:-3.010, Mutant3:-2.942, Mutant 4:-3.542, and the entire results are in 'result.csv’. Visualization are shown on the right.

Involved roles:
AI-PM
Two AI-DE (one for inverse folding and the other for visualization)
AI-Pr (multimodal visualization)

Fig. 2. Three representative use case examples of the ProteinEngine in user mode, where only the absence of mandatory parameters will be requested to the user.

most of the task scenarios in protein design. Each category is mapped to a specific AI-DE. To provide a tangible understanding of the ProteinEngine in action, we illustrate its use through distinct case examples in Fig. 2, where different AI-DEs are involved in each case. We illustrate typical examples of SOTA AI models in protein design which have been used in our ProteinEngine platform.

3 User Study

Hypothesis Formulation and Testing. To evaluate the effectiveness of ProteinEngine, we conducted a user study focused on gauging its proficiency as an intuitive, human-centered system for protein engineering tasks. We employed hypothesis testing to quantitatively compare the performance of ProteinEngine against a baseline condition, focusing on task completion time, number of attempts, system usability, and the perceived workload. In the baseline condition, participants employed traditional tools and methods, independent of ProteinEngine, such as executing Python scripts directly. Therefore, our null hypotheses were formulated as follows:

(H1) ProteinEngine does not reduce the time required for successful identification and execution of protein engineering models against the baseline.
(H2) ProteinEngine does not improve the accuracy in identifying and executing models for protein engineering tasks against ProteinEngine does improve the accuracy against the baseline.
(H3) ProteinEngine does not enhance the overall system usability for model identification and execution within protein engineering tasks against the baseline.
(H4) ProteinEngine does not decrease the workload required for the completion of protein engineering model identification and execution against the baseline.

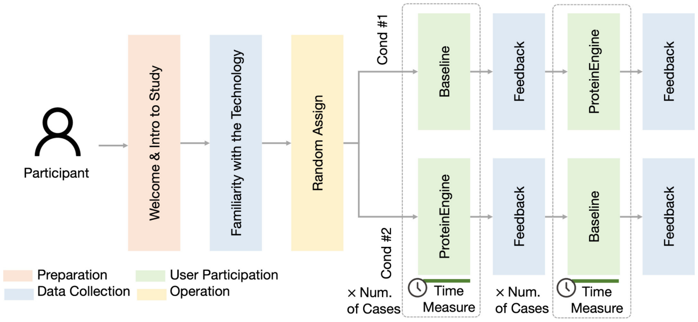


Fig. 3. The overall flowchart of the user study. This includes preparation (participant recruitment and briefing), user participation operation (familiarization with the technology and random assignment to conditions), data collection (sequential tasks under different conditions with intermittent feedback). Each stage of the process is color-coded for ease of understanding. (Color figure online)

Based on the hypotheses, we used a single-sided t-test to assess statistical significance.

Dependent Variables. Statistical tests on these hypotheses involve collecting data on the dependent variables from the user study, specifically the task completion time, number of attempts, usability score, and workload index, which are defined as follows.

- Task Completion Time: This objective, continuous variable measures the total time each participant takes to successfully complete a task under each condition.
- Number of Attempts: Another objective, continuous variable records the total attempts a participant takes to successfully complete each task under each condition. This variable is indicative of the accuracy of user actions.
- Usability Score: This subjective, continuous variable is derived from the System Usability Scale (SUS) questionnaire [2]. The score reflects the perceived usability of the system.
- Workload Index: This subjective, continuous variable, sourced from the NASA Task Load Index (NASA TLX) questionnaire [8], assesses the perceived mental workload across six dimensions: mental demand, physical demand, temporal demand, effort, performance, and frustration level.

Independent Variables. The primary independent variable is the Condition under which participants perform the protein engineering tasks, either the baseline or the ProteinEngine. Additionally, we consider potential confounding independent variables that could influence our study outcomes, including:

- Participant Background: These categorical variables encapsulate information about each participant's professional role and affiliations (academia or industry). This information could offer insights into a user's likely background knowledge and potential biases or preferences when using the interface.
- Familiarity with Technology: These numerical variables represent the degree of each participant's familiarity with protein engineering tasks, Python programming language, AI models, and the intersection of these areas. The level of familiarity could potentially influence the ease with which participants adapt to the ProteinEngine, and thus might impact the measurements of variables like task completion time, number of attempts, and perceived usability and workload.

User Study Design. The user study workflow, shown in Fig. 3, consisted of several steps. First, participants received an introductory tutorial that provided information about the study's background, motivation, and procedures. Next, participants completed a preliminary questionnaire that assessed their background knowledge and familiarity with AI, protein engineering, and their interdisciplinary overlap. The study followed a between-subjects design, comparing participants exposed to two conditions: a baseline condition and the ProteinEngine condition. To mitigate learning effects, the order in which participants encountered these conditions was randomized. To control for potential effects stemming from participants' background and familiarity with technology, all participants completed the same set of activities under both conditions (baseline and ProteinEngine). This approach provided paired data for analysis. Participants were assigned a series of six distinct protein engineering tasks under each experimental condition. These tasks included protein folding, inverse protein folding, and protein mutation prediction. During the task completion process, we carefully recorded the total time taken for each task and the number of attempts required for successful execution. After completing the tasks in either the baseline or ProteinEngine condition, participants were asked to fill out a questionnaire. This questionnaire aimed to assess their subjective impressions of the system's usability and their perceived workload during task completion. To ensure a valid comparison of user experiences, identical questionnaires were administered. Data was collected using Google Sheets, with all questions being mandatory to prevent missing data. Incomplete data from participants who withdrew or failed to complete tasks were excluded from the analysis.

Participants Recruitment. We strategically planned the recruitment of volunteer participants to encompass a wide range of potential users. This design aimed to test the versatility and broad applicability of our proposed method across various user groups. Our participant cohort consisted of volunteers from both the AI and biological communities, spanning both academic and industrial fields. This diverse group, including students, AI researchers, lab technicians, and biologists, allows for a comprehensive evaluation of ProteinEngine's functionality across multiple user profiles.

Implementations. For the LLM in ProteinEngine, we chose the gpt-3.5-turbo. As the most advanced model in the GPT-3.5 series, this version provides robust capabilities and superior performance suitable for our application.

Collected Data. The boxplots illustrating the distribution of our four key variables, namely Task Completion Time, Number of Attempts, Usability Score, and Workload Index, can be found in Fig. 4. Each box plot provides a visual summary of the minimum, first quartile (Q1), median (Q2), third quartile (Q3), and maximum values for these variables. The box represents the interquartile range (IQR) from Q1 to Q3, the line inside the box denotes the median, and the whiskers extend to show the range of the data within 1.5 times the IQR. Observations beyond this range are considered outliers and are represented as individual points. On average it take 36.40 min to complete the user study for each participant.

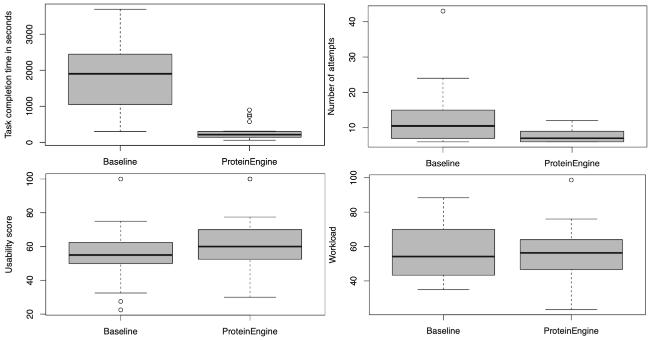


Fig. 4. Boxplots for the four variables.

Table 2. Null hypothesis testing results.

Hypothesis	Observed test statistics	P-value	Reject null hypothesis
H1	7.7012	2.335×10^{-8}	Yes
H2	3.1944	1.884×10^{-3}	Yes
H3	−2.4162	1.166×10^{-2}	Yes
H4	0.74029	2.330×10^{-1}	No

Hypothesis Testing. We applied the hypothesis testing to ascertain whether the performance differences observed between the two conditions i.e., baseline and ProteinEngine were statistically significant. The differences under consideration, represented as d_i, were computed by subtracting the ProteinEngine measurements from the baseline measurements. Under the null hypothesis, where both platforms have an equivalent effect, these differences should follow a distribution centered around zero, i.e., $\mu_d = 0$. Our final dataset for hypothesis testing comprised $n = 26$ samples. We formulated our null and alternative hypotheses as follows: H_0: $\mu_d = 0$ against H_1: $\mu_d > 0$. This holds for testing hypotheses H1, H2, and H4. For testing H3, the alternative hypothesis is $\mu_d < 0$.

Let $\overline{\mathrm{d}}$ and sd represent the sample mean and sample standard deviation of the observed differences, respectively. Given these parameters, the sampling distribution for the test statistic follows a t distribution with degrees of freedom $\mathrm{n} - 1$. This means that, under null hypothesis H_0, $\tau = \frac{\overline{\mathrm{d}}}{\mathrm{s_d}/\sqrt{\mathrm{n}}} \sim \mathrm{t}_{\mathrm{n}-1}$. Table 2 compiles our hypothesis testing results. Of note, no significant difference was found for the workload measure between the two conditions.

Results. Our study encompassed a total of 45 participants. For hypothesis testing, we employed the paired two-sample t-test (effectively a one-sample, one-sided t-test on the difference) at a 5% significance level for the four variables under two conditions, baseline and ProteinEngine. The results allowed us to reject three of the null hypotheses, thereby highlighting the superior performance of ProteinEngine in facilitating protein engineering tasks.

4 Conclusion

In this work, we presented ProteinEngine, a groundbreaking platform that amplifies the capabilities of LLMs in the realm of protein engineering. This platform's human-centered design greatly eases the learning curve traditionally associated with specialized tools, thereby making protein engineering tasks more accessible. By integrating advanced LLMs with domain-specific expertise, ProteinEngine marks a significant leap forward in the application of AI to protein engineering, showing great potential to accelerate scientific discoveries and spur innovation. As we continue to develop and refine ProteinEngine, it is critical to emphasize the importance of responsible use and rigorous validation. Therefore, the development of comprehensive ethical guidelines and robust validation protocols is a key direction for future work.

References

1. Bileschi, M.L., et al.: Using deep learning to annotate the protein universe. Nat. Biotechnol. **40**(6), 932–937 (2022)
2. Brooke, J., et al.: SUS-A quick and dirty usability scale. Usability Eval. Ind. **189**(194), 4–7 (1996)
3. Corso, G., et al.: DiffDock: diffusion steps, twists, and turns for molecular docking. In: The Eleventh International Conference on Learning Representations (2023)
4. Dong, Q., et al.: A survey for in-context learning. arXiv preprint arXiv:2301.00234 (2022)
5. Evans, R., et al.: Protein complex prediction with AlphaFold-Multimer. bioRxiv, pp. 2021–10 (2022)
6. Ferruz, N., et al.: ProtGPT2 is a deep unsupervised language model for protein design. Nat. Commun. **13**(1), 4348 (2022)
7. Ganea, O.-E., et al.: Independent SE(3)-equivariant models for end-to-end rigid protein docking. In: International Conference on Learning Representations (2021)
8. Hart, S.G., et al.: Development of NASA-TLX (task load index): results of empirical and theoretical research. Adv. Psychol. **52**, 139–183 (1988)
9. Jumper, J., et al.: Highly accurate protein structure prediction with AlphaFold. Nature **596**(7873), 583–589 (2021)

10. Khurana, S., et al.: DeepSol: a deep learning framework for sequence-based protein solubility prediction. Bioinformatics **34**(15), 2605–2613 (2018)
11. Lin, Z., et al.: Evolutionary-scale prediction of atomic-level protein structure with a language model. Science **379**(6637), 1123–1130 (2023)
12. Luo, S., et al.: Antigen-specific antibody design and optimization with diffusion-based generative models for protein structures. In: Oh, A.H., Agarwal, A., Belgrave, D., Cho, K. (eds.) Advances in Neural Information Processing Systems (2022)
13. Madani, A., et al.: Large language models generate functional protein sequences across diverse families. Nat. Biotechnol., 1–8 (2023)
14. Rao, R.M., et al.: MSA transformer. In: Meila, M., Zhang, T. (eds.) Proceedings of the 38th International Conference on Machine Learning, volume 139 of Proceedings of Machine Learning Research, pp. 8844–8856. PMLR, 18–24 July 2021
15. Stärk, H., et al.: EquiBind: geometric deep learning for drug binding structure prediction. In: International Conference on Machine Learning, pp. 20503–20521. PMLR (2022)
16. Yi, K., et al.: Graph denoising diffusion for inverse protein folding. In: Advances in Neural Information Processing Systems, 36 (2024)
17. Zhang, Z., et al.: Protein representation learning by geometric structure pretraining. In: The Eleventh International Conference on Learning Representations (2023)
18. Zhao, W.X., et al.: A survey of large language models. arXiv preprint arXiv:2303.18223 (2023)
19. Zhou, B., et al.: Accurate and definite mutational effect prediction with lightweight equivariant graph neural networks. arXiv preprint arXiv:2304.08299 (2023)

Wearable Devices, Sensors, and Robotics

Advancements in Non-invasive AI-Powered Glucose Monitoring: Leveraging Multispectral Imaging Across Diverse Wavelengths

Tahsin Kazi[1], John Oakley[2], Anh Duong[1], El Arbi Belfasi[1],
Katherine Ingram[1], and Maria Valero[3](✉)

[1] Department of Computer Science, College of Computing and Software Engineering, Kennesaw State University, Marietta, GA, USA
[2] Department of Exercise Science and Sport Management, Wellstar College of Health and Human Services, Kennesaw State University, Kennesaw, GA, USA
[3] Department of Information Technology, College of Computing and Software Engineering, Kennesaw State University, Marietta, GA, USA
mvalero2@kennesaw.edu

Abstract. The pursuit of non-invasive glucose monitoring has leveraged multispectral imaging technology. Our study focuses on improving GlucoCheck, a non-invasive AI-powered glucose monitor. Through processing a dataset of 3600 images, we derived four subsets for unique wavelengths and trained regression models. Performance evaluations, employing Mean Absolute Error (MAE), Clarke Error Grids (CEG), and Bland-Altman Plots (BAP), revealed promising outcomes, with a median MAE of 2.06 mg/dl, CEG Zone A% of 99.17%, and percentage of BAP outliers as 3.66%. While no clear correlation was found between wavelength and statistical accuracy, a significant relationship emerged between wavelength and clinical accuracy through agreement with BAP outliers. These nuanced findings highlight the potential of multispectral imaging and machine learning in advancing accurate glucose estimation.

Keywords: Glucose Monitoring · Multispectral Imaging · Wavelength · Machine Learning · Biosensor

1 Introduction

Diabetes, a chronic metabolic disorder impacting over 500 million adults globally [1], has been characterized as the epidemic of the century [2]. The surge in this metabolic ailment, alongside others, has spurred interest in continuous glucose monitoring (CGM), which has shown potential in diabetes remission, dosage reduction, and improving patient well-being [3]. Existing glucose monitoring techniques encompass both invasive and non-invasive modalities [4].

Typical invasive techniques encompass blood sample analysis, finger-prick glucometers, and CGMs employing subcutaneous needles [5]. Non-invasive glucose monitoring methods circumvent tissue interference through approaches like

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 387–396, 2024.
https://doi.org/10.1007/978-3-031-66538-7_38

optical spectroscopy, photoacoustic spectroscopy, electromagnetic sensing, and nanomaterial-based sensing [5,6]. Optical spectroscopy, among the foremost non-invasive methods, utilizes diverse light sources to gauge absorption, reflection, and transmission [7].

Near-Infrared (NIR) spectroscopy, a derivative of this method, employs light absorption within the near-infrared spectrum. By measuring and analyzing the penetration and absorption patterns of NIR light in human tissue, it enables the estimation of blood glucose concentration. This technique presents a safe, painless, and convenient means for continuous glucose monitoring.

Related works using this technology have already shown promising results. The device in [8] used NIR spectroscopy with PPG processing and regression to estimate glucose values for 75 subjects and achieved 82% clinical accuracy. Another model [9] used Mid-Infrared Spectroscopy with 3 distinct wavelengths and Multiple Linear Regression to achieve 86.3% clinical accuracy. A similar method in [10] used analog NIR spectroscopy, boosted by fuzzy logic to create a model with 97.5% clinical accuracy. For other optical-spectroscopy-based work, refer to [4,7,11,12].

Although these works present nuanced implications, they are yet to explore hardware factors such as light wavelength. To tackle this issue, we are evaluating the effect of various wavelengths on the performance of our non-invasive blood glucose monitoring system, GlucoCheck [13,14]. Through this investigation, we hope to determine the most effective approach for NIR spectroscopy-based blood glucose monitoring. This paper will present an experimental evaluation of different wavelengths, a comparative analysis, and a novel non-invasive CGM device for glucose estimation.

2 Previous Implementation

In our initial design (Fig. 1), an NIR laser coupled with a finger-clip was employed to capture images via a camera. These images were then utilized to extract features, subsequently employed in training a machine-learning model for glucose estimation. Specifically, our prototype utilized an Arducam 5MP camera diode alongside a KY-008 650 nm laser diode for operation [13]. Through model training, we found that the KNeighbors regressor, boosted with AdaBoost, yielded the most favorable outcomes: an MAE of 9.4 mg/dl and 90.78% CEG Zone A% [14]. These findings serve as a cornerstone for our ongoing research endeavors.

3 Methodology

3.1 Research Objective

The objective is to understand the effect of various wavelengths on the GlucoCheck device's performance in estimating blood glucose levels. The research involves building a new device to simulate diverse wavelengths and evaluate their effects on accuracy, efficiency, and functionality.

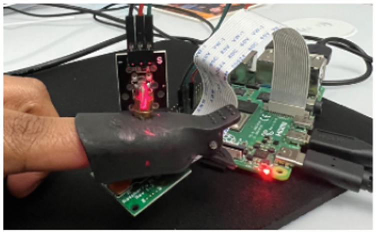


Fig. 1. Previous GlucoCheck Prototype

3.2 Wavelength Selection

The wavelength selection process considered hardware compatibility, affordability, and physiological impact on human subjects. In this study, we carefully selected four wavelengths (650 nm, 808 nm, 830 nm, 850 nm) as they were affordable and compatible with the Raspberry Pi GPIO board. Importantly, prior studies leveraging similar wavelengths have successfully estimated glucose levels [15]. Figure 2 shows the four laser diodes selected.

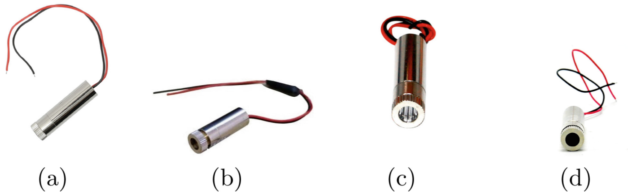


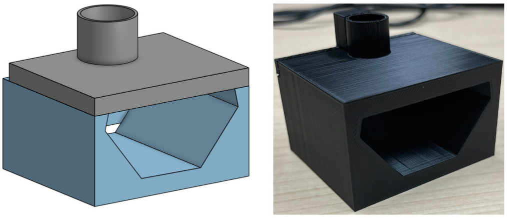
Fig. 2. (a) Laser 650 nm, (b) Laser 808 nm, (c) Laser 830 nm, and (d) Laser 850 nm.

In evaluating safety, we considered Maximum Permissble Exposure (MPE), measuring the total radiant light exposure produced by the lasers [16]. The MPE limit for human skin is 1.0 W/cm^2 [17] and all diodes fell below this value (650 nm: 0.231 W/cm^2, 808 nm: 0.375 W/cm^2, 830 nm: 0.425 W/cm^2, 850 nm: 0.426 W/cm^2), confirming their safety for use on the skin.

3.3 Camera Selection and Enclosure Creation

The proposed device uses the same PiCamera 2 camera-diode and Raspberry Pi4 Model B from the previous implementation due to their extensibility and compatibility with new hardware. Although prior studies [18] favored lower-definition (640 p) images for faster processing, a reassessment explored high-definition (1080 p) image capture. While the previous implementation used a fingerclip to house the laser and camera, we are implementing a new box-shaped enclosure for this experiment. The larger, more open box shape was chosen to account for laser and camera diodes of varying sizes and specifications. In addition to housing other components, the enclosure must also block out light that could interfere with data collection.

To design this enclosure, we utilized a computer-aided design (CAD) program, Onshape, to create a 3D part model and then manufactured it using black PLA filament on the Creality Ender 3 printer. Within our design, the laser is positioned within the chimney atop the box and the camera is slotted through the back crevice. Figure 3 shows the initial CAD design and the final printed model.

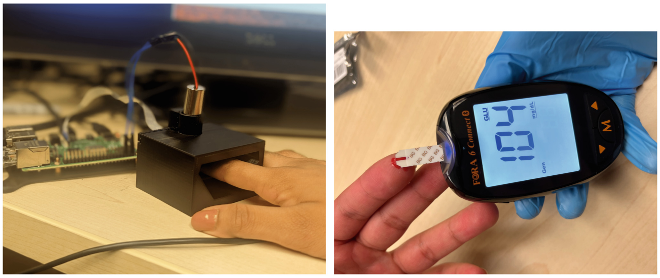


(a) (b)

Fig. 3. New Enclosure. (a) Onshape Design. (b) Final Printed Model.

3.4 Data Collection

Upon receiving IRB approval, data collection started with 25 participants, encompassing various skin tones and ages. The clinical characteristics of the participants are gender (14 females and 11 males), ethnicity (8 Caucasians, 6 Latinos, 5 African American, 6 Asian), and all were non-diabetic patients. Across the 25 participants, a total of 30 sets of data were gathered, each set including images for each wavelength and a reference glucose value. The reference values were obtained with a glucometer, which ranged from 78 mg/dl to 165 mg/dl. Our procedure involved placing the patient's finger inside the front entry of the



(a) (b)

Fig. 4. Data collection of a participant of the study. (a) Using GlucoCheck. (b) Using Fora 6 glucometer.

enclosure, directly over the PiCamera. Then the laser is placed into the enclosure's chimney and the data collection script was run on the Raspberry Pi. A live video feed displayed the images being captured for the model's evaluation. After the script is finished, the laser is replaced with another until all four lasers have been used. Figure 4 illustrates the data collection process.

4 Experiments and Results

4.1 Data Cleaning and Feature Extraction

The initial dataset comprised images stored in folders, with information on the wavelength, participant ID, and reference glucose value. Post-analysis and the removal of poorly captured images, the finalized collection comprised 3600 high-quality images, with 900 images per wavelength.

Employing the same approach from our previous study [14], this dataset underwent extensive processing, extracting RGB intensity values and statistical measurements from images. Subsequently, a refined dataset emerged, comprising 3600 samples, each linked to 290 features: 256 features for red intensity values, 31 measurement features, 1 reference glucose value, 1 wavelength number (650, 808, 830, or 850), and 1 subject ID. The intensity features map the frequency of specific RGB values for the red color channel in the image while the measurement features map statistical measures, such as mean, median, and mode, of all RGB values in the image. Further explanation can be found in our previous work [14].

4.2 Model Training

The processed dataset was split into four smaller subsets, each dedicated to a specific wavelength. Subsequently, each wavelength dataset was used to create a training/testing split. With a randomized shuffle of the samples, the data splits were arranged with 60% training data and 40% testing data. The training and testing splits contained the all 290 features except for the wavelength and subject ID. These splits were used to train four KNeighbours regressors with AdaBoost ensemble learning following the same procedure and hyperparameters as our last study [14].

4.3 Performance Metrics

While the task of estimating blood glucose levels is primarily a regression analysis problem, a crucial secondary task is validating a novel medical instrument. This nuance led us to choose performance metrics that consider machine-learning and medical perspectives. Three metrics were selected: MAE, CEG, and BAP. These metrics, respectively, assess the models' statistical accuracy, clinical accuracy through correlation, and clinical accuracy through agreement. MAE is a standard measure of accuracy in regression analysis. This metric is calculated by averaging the difference of all predicted and reference blood glucose values. This metric is recorded in mg/dl and accounts for raw statistical accuracy. To account for clinical accuracy, however, two approaches must be considered: correlation and agreement.

Correlation examines the proportionality of the model's predictions to the reference values, represented by CEG in this study. These grids categorize the relationship between predicted and reference blood glucose values into five distinct zones, each delineating various levels of accuracy and clinical implications from incorrect measurement [19]. Our assessment recorded the grid for each model and the percentage of predictions falling within Zone A, denoted as A in Fig. 5.

Correlation is robust but assumes comparison between two distinct variables, which may not always hold true. To address this, agreement should be considered, examining relationships between variables measuring the same attributes using different methods. Bland-Altman Plots, a standard measure for agreement in medical/chemical instruments, display observed and predicted values' mean on the x-axis and their difference on the y-axis [20]. They provide insights into error distribution and density, establishing thresholds for insignificant model inaccuracies. We utilized the 95% range of the normal distribution to set upper and lower error limits, identifying outliers as predictions outside these bounds. Although all three metrics are valuable, discussion will focus more heavily on CEG and BAP as they evaluate the device's medical efficacy, which is more important than statistical accuracy.

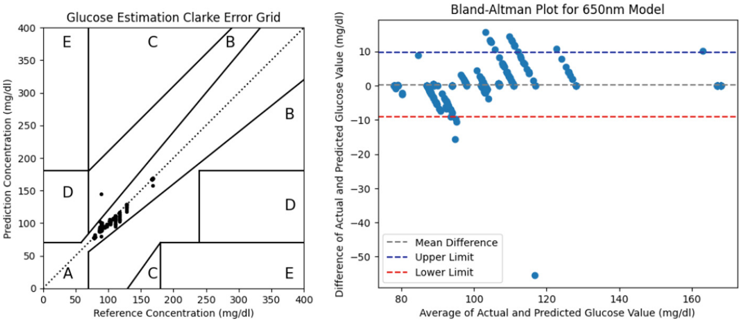
4.4 Experimental Results

After conducting tests on the models using their respective testing splits, key metrics such as MAE in mg/dl, Zone A%, and the percentage of BAP outliers were recorded and tabulated in Table 1. The CEG and the BAP were plotted for each wavelength to visualize the performance, however, we will only show the first for reference as it was the best-performing model. Figure 5 shows the CEG and BAP for the 650 nm wavelength model.

For further analysis, we plotted each metric against wavelength, as illustrated in Fig. 6, to delineate a relationship between wavelength and the given metric. The values present on the plots are gathered from Table 1. Furthermore, linear regression equations were fitted onto each scatterplot using the least-squares method, as it provides the best estimation.

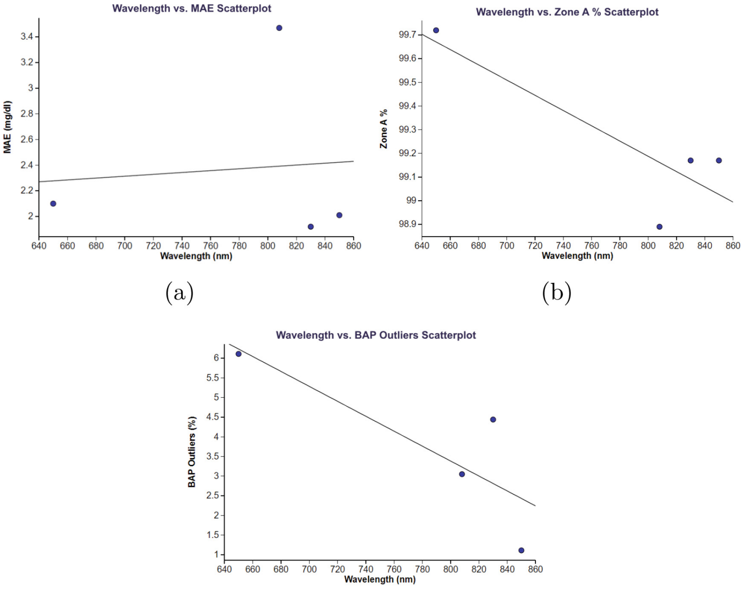
Table 1. Performance Metrics for All Wavelength Models

Wavelength	MAE	Zone A %	BAP Outliers
650 nm	2.10 mg/dl	99.72%	6.11%
808 nm	3.47 mg/dl	98.89%	3.05%
830 nm	1.92 mg/dl	99.17%	4.44%
850 nm	2.01 mg/dl	99.17%	1.11%



(a) (b)

Fig. 5. Results with laser 650 nm. (a) CGE. (b) BAP.



(a) (b)

(c)

Fig. 6. Scatterplot of wavelength against assessment metrics. (a) wavelength vs. Zone A % (b) wavelength vs. MAE. (c) wavelength vs. BAP outliers.

To analyze the impact of wavelength on model performance, the three performance areas from experimentation were considered: statistical accuracy, clinical accuracy via correlation, and clinical accuracy through agreement. These areas not only represent the chosen metrics, but also encompass the validity of the GlucoCheck device in machine learning, glucose estimation, and clinical efficacy. Subsequent sections will present an in-depth analysis of each facet.

4.5 Statistical Accuracy

Statistical accuracy, gauged by MAE, reveals pertinent distinctions between the wavelengths. The 808 nm model showcases the highest MAE, significantly surpassing the other wavelengths, which all had similar values. This difference points to the 808 nm model's MAE being a statistical outlier. The interquartile range for MAE is 1.92 mg/dl and a straightforward outlier test highlights the 808 nm model's MAE (3.47 mg/dl) as surpassing the standard threshold $(Median + (1.5 * IQR))$, categorizing it as an outlier. These findings suggest no clear correlation between wavelength and statistical accuracy, with consistent accuracy across wavelengths but skewed variance from an outlier, further supported by the small correlation coefficient between MAE and wavelength $(r = 0.09)$.

4.6 Clinical Accuracy Through Correlation

Clinical accuracy, shown by CEGs, aligns closely with statistical accuracy. Zone A % exhibits minimal variance across the four wavelengths, hovering around the median value of 99.17% with a standard deviation of 0.37%. Despite a high negative correlation between Zone A% and wavelength ($r = -0.84$), the negligible variance renders this correlation less impactful. However, a notable outlier emerges in the 808 nm model again, with its higher error and larger MAE compared to other wavelengths. This alignment underscores a similarity between statistical accuracy and clinical accuracy through correlation.

4.7 Clinical Accuracy Through Agreement

Finally, we assess clinical accuracy through agreement, employing BAPs and outlier percentages. Here, a pertinent difference between the lowest and highest wavelengths emerges with the 650 nm model's 6.11% outlier rate and the 850 nm model's 1.11% outlier. Despite a median outlier percentage of 3.66%, the noticeable standard deviation of 2.12% implies variance by surpassing half of the median's value. Furthermore, a strong negative correlation between outlier percentage and wavelength ($\mathrm{r} = -0.81$) suggests a potential relationship between wavelength and clinical accuracy through agreement. This could suggest that higher wavelengths induce more agreement, which is reinforced by the BAP, demonstrating fewer outliers in the healthy blood glucose range (70 mg/dl to 100 mg/dl) for higher wavelengths.

5 Discussion

Through our analysis, we find no association between wavelength, statistical accuracy, and clinical accuracy through correlation. Although a relationship between wavelength and clinical accuracy through agreement is evident, it only encompasses one area of evaluation. A definitive conclusion remains elusive. While our findings echo fundamental attributes of blood glucose and spectrometry, such as glucose absorption peaking in much higher wavelengths [21]; substantial improvements in model accuracy are contingent upon a considerable increase in wavelength. To surmount this limitation, a subsequent study intends to delve into a much wider range of wavelengths and revisit the association between wavelength and performance.

6 Conclusion

In conclusion, this study marks a significant stride toward refining non-invasive glucose monitoring methodologies by leveraging multispectral imaging and machine learning. The exploration across four distinct wavelengths has showcased promising outcomes in glucose estimation. Notably, the models exhibited substantial accuracy, with minimal MAE values ranging from 1.92 mg/dl to 3.47 mg/dl across the wavelengths. The CEG analysis revealed high percentages of predictions within Zone A, with percentages exceeding 98.89% across all wavelengths. Additionally, the BAP analysis identified a lower percentage of outliers, particularly in the healthy glucose range (70 mg/dl to 100 mg/dl) for higher wavelengths. While no association emerged between wavelength, statistical accuracy, and clinical accuracy through correlation, a potential association with clinical accuracy through agreement warrants further exploration. These findings underscore the promise of multispectral imaging and machine learning in advancing glucose estimation, urging further research in broader wavelength ranges for improved biosensing applications.

Acknowledgments. Research reported in this publication was supported by the National Institute On Aging of the National Institutes of Health under Award Number P30AG073105.

References

1. Sapra, A., Vaqar, S., Bhandari, P.: Diabetes Mellitus -PMID 31855345, 12 (2019)
2. Kharroubi, A., Darwish, H.: Diabetes mellitus: the epidemic of the century. World J. Diab. **6**(6), 850–867 (2015)
3. Srivastava, S.B.: Empowering people with diabetes: role of continuous glucose monitor systems. Am. J. Lifestyle Med. **17**(3), 359–364 (2023)
4. Tang, L., Chang, S.J., Chen, C.-J., Liu, J.-T.: Non-invasive blood glucose monitoring technology: a review. Sensors **20**(23), 6925 (2020)

5. Villena Gonzales, W., Mobashsher, A., Abbosh, A.: The progress of glucose monitoring-a review of invasive to minimally and non-invasive techniques, devices and sensors. Sensors **19**(4), 800 (2019)
6. Hina, A., Saadeh, W.: Noninvasive blood glucose monitoring systems using near-infrared technology-a review. Sensors **22**, 4855 (2022)
7. Shokrekhodaei, M., Quinones, S.: Review of non-invasive glucose sensing techniques: optical, electrical and breath acetone. Sensors **20**(5), 1251 (2020)
8. Reddy, P., Mahesh, D., Teja, C., Janaki, M., Mannem, K.: Non-invasive glucose monitoring using NIR spectroscopy. J. Phys. Conf. Ser. vol. 2325, p. 012021 (2022)
9. Kasahara, R., Kino, S., Soyama, S., Matsuura, Y.: Noninvasive glucose monitoring using mid-infrared absorption spectroscopy based on a few wavenumbers. Biomed. Opt. Exp. **9**(1), 289 (2018)
10. Darwich, M.A., Shahen, A., Daoud, A., Lahia, A., Diab, J., Ismaiel, E.: Non-invasive IR-based measurement of human blood glucose. Eng. Proc. **35**(1) (2023)
11. Hina, A., Saadeh, W.: Noninvasive blood glucose monitoring systems using near-infrared technology—a review. Sensors **22**(13) (2022)
12. Alsunaidi, B., Althobaiti, M., Tamal, M., Albaker, W., Al-Naib, I.: A review of non-invasive optical systems for continuous blood glucose monitoring. Sensors **21**, 6820 (2021)
13. Valero, M., et al.: Development of a noninvasive blood glucose monitoring system prototype: pilot study. JMIR Formative Res. **6**(8), e38664 (2022)
14. Kazi, T., Ponakaladinne, K., Valero, M., Zhao, L., Shahriar, H., Ingram, K.H.: Comparative study of machine learning methods on spectroscopy images for blood glucose estimation. In: International Conference on Pervasive Computing Technologies for Healthcare, pp. 60–74, Springer (2022). https://doi.org/10.1007/978-3-031-34586-9_5
15. Monte-Moreno, E.: Non-invasive estimate of blood glucose and blood pressure from a photoplethysmograph by means of machine learning techniques. Artif. Intell. Med. **53**(2), 127–138 (2011)
16. Thomas, R.J., Rockwell, B.A., Marshall, W.J., Aldrich, R.C., Zimmerman, S.A., Rockwell, R.J., Jr.: A procedure for multiple-pulse maximum permissible exposure determination under the z136. 1–2000 American national standard for safe use of lasers. J. Laser Appl. **13**(4), 134–140 (2001)
17. Yakupoglu, S., Hakan, T.: Laboratory safety handbook. https://fens.sabanciuniv.edu/sites/fens.sabanciuniv.edu/files/2021-08/labsafety_web.pdf. Accessed 03 Sep 2023
18. Valero, M., Ingram, K.H., Duong, A., Nino, V.: Pervasive glucose monitoring: a non-invasive approach based on near-infrared spectroscopy. In: 2023 International Conference on Pervasive Computing Technologies for Healthcare, Springer, Forthcoming. https://doi.org/10.1007/978-3-031-59717-6_19
19. Clarke, W.L., Cox, D., Gonder-Frederick, L.A., Carter, W., Pohl, S.L.: Evaluating clinical accuracy of systems for self-monitoring of blood glucose. **10**(5), 622–628 (1987)
20. Bland, J.M., Altman, D.G.: Statistical methods for assessing agreement between two methods of clinical measurement. Lancet **1**, 307–310 (1986)
21. Tang, J.-Y., Chen, N.-Y., Chen, M.-K., Wang, M.-H., Jang, L.-S.: Dual-wavelength optical fluidic glucose sensor using time series analysis of d(+)-glucose measurement. Jpn. J. Appl. Phys. **55**, 106601 (2016)

Anticipating Stress: Harnessing Biomarker Signals from a Wrist-Worn Device for Early Prediction

Marina Andrić[1(✉)], Mauro Dragoni[1], and Francesco Ricci[2]

[1] Fondazione Bruno Kessler, Trento, Italy
{mandric,dragoni}@fbk.eu
[2] Free University of Bozen-Bolzano, Bolzano, Italy
fmr959@gmail.com, fricci@unibz.it

Abstract. Stress acts as a triggering and aggravating factor for many diseases and health conditions. This has prompted the development of wearable devices capable of continuously and unobtrusively tracking physiological signals associated with stress levels. Moreover, data mining methods have been devised to extract valuable information from these signals, to detect and monitor stress more effectively. We argue that it is possible to accurately detect and differentiate physiological changes occurring at the early onset of stress, i.e., the anticipation stage, from those occurring in no-stress, stress, and post-stress conditions. To investigate it, we analyze biomarker data (blood volume pulse, skin conductance, skin temperature, and acceleration) collected from wrist sensors in two publicly available datasets, where psychosocial stress is induced under controlled laboratory conditions. We train and evaluate person-specific classification algorithms by using established learning approaches. We have discovered that the random forest classifier yields promising results in both detecting stress anticipation and distinguishing between the four considered classes. The results of this study suggest that wearable systems, incorporating sensors and stress monitoring algorithms like the ones introduced here, can become integral components of intervention systems aimed at addressing stress-related issues.

1 Introduction

Stress is a widely discussed topic and plays a central role in human life. Its increased coverage in media has led to a rise in research and public awareness about its effects. It is now well-established that stress can affect health directly through autonomic and neuroendocrine responses [17], as well as indirectly by influencing health behaviours [25]. Stressors provoke a range of physiological responses that are initially protective and adaptive, yet when the balance is disrupted, and the response persists over time, it can become detrimental to an individual's health and well-being [16]. Stress may indirectly contribute to conditions such as obesity, cardiovascular disease, and cancer risk, as it can lead to negative changes in diet and perpetuate unhealthy eating behaviors [27].

In response to increased global mental health concerns in 2020, the World Health Organization (WHO) released a guide aimed at providing individuals

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 397–408, 2024.
https://doi.org/10.1007/978-3-031-66538-7_39

with practical stress coping strategies [18]. These include ensuring sufficient sleep, monitoring emotions, prioritizing tasks, identifying stressors, practicing daily muscle relaxation, and cultivating inner contentment and peace. Research has assessed stress management strategies, especially their impact on university students [5]. Findings show targeted interventions effectively reduce stress within this group [4]. Additionally, a clinical trial found a mobile app mindfulness program to significantly reduced perceived stress levels among students [22].

Stressors trigger a range of biological responses involving the autonomic nervous system, immune system, and hypothalamic-pituitary-adrenal axis, which can be quantified using biomarkers including cortisol response magnitude, heart rate, electrodermal activity, and heart rate variability [7]. The stress response is not only evoked by experiencing a stressful event but also by anticipating its onset [24]. This anticipation prompts individuals to assess the perceived threat to their well-being through cognitive evaluation. In this assessment, psychological factors such as novelty, unpredictability, social evaluation, and a sense of low control have been identified as contributors to the stress response [9].

Recent advances in wearable technology have facilitated the development of unobtrusive, user-friendly devices that are capable of continuously recording multiple robust indicators of elevated levels of stress [19]. With the growing demand for stress management solutions, consumer wearables, such as wrist-worn and head-worn variants, have emerged as popular choices in recent years. Leading wrist-worn devices like those from Garmin and Fitbit leverage signals, such as heart rate variability and electrodermal activity, to compute real-time and daily stress scores [2,10].

In this study, we investigate the potential for detecting different types of physiological changes linked to stress, focusing on stress anticipation, i.e., the state preceding actual stress, using biomarkers that can be monitored comfortably by a wrist-worn device.

1.1 Motivation and Contribution

The existing research literature on stress monitoring using wearable devices has predominantly approached stress prediction as a binary condition. Typically, datasets containing stress biomarkers are labeled with binary stressed or non-stressed time periods, and models trained on these datasets yielded classifiers capable of classifying previously recorded data as either stressed or non-stressed [28]. Intending to forecast future stress episodes, Umematsu et al. [26] focused on predicting tomorrow's stress score from physiological data from wrist-worn devices collected the previous day. Similarly, the work of Jaimes et al. [14] explored the use of physiological time series data collected by a body sensor network to predict future stress episodes.

We test the hypothesis that stress anticipation associated with social evaluation can be accurately predicted using biomarker data collected from a wrist-worn device. Furthermore, we propose that physiological changes occurring during stress anticipation can be differentiated among no stress, stress, and post-stress conditions, framing this as a four-class classification problem. To investigate this hypothesis, we analyze two publicly available datasets: WESAD [20] and Stress-Predict [12]. These datasets feature the Trier Social Stress Test (TSST), a psychosocial test conducted in controlled laboratory settings. Based on their study designs, we define stress anticipation as approximately seven minutes before the stress condition, and post-stress as around four minutes following the stress condition. Recognizing the diverse individual responses to stress in identical situations [21], we develop person-specific prediction models, i.e., models trained only with data pertaining to the considered subject.

2 Methodology

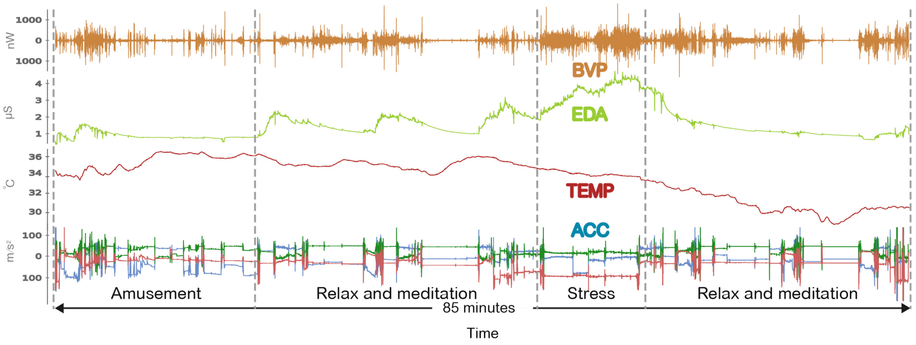


Fig. 1. Data from a single subject with ID 5 from the WESAD dataset, excluding baseline monitoring at the beginning.

Both the WESAD and Stress-Predict datasets were derived from biomarker data collected using the Empatica E4 device [1], capturing measurements of blood volume pulse (BVP), which tracks changes in blood volume within blood vessels over time, electrodermal activity (EDA), skin temperature (TEMP), and three-axial acceleration (ACC). The E4 records BVP at 64 Hz, EDA at 4 Hz, TEMP at 4 Hz, and ACC at 32 Hz.

Both experimental setups incorporate the Trier Social Stress Test, a well-established protocol known to elicit responses from the hypothalamic-pituitary-adrenal axis [3]. The TSST induces psychosocial stress by requiring participants to deliver an interview-style presentation, followed by a challenging mental arithmetic task conducted in front of an audience that provides no feedback or encouragement. In addition to the TSST, the Stress-Predict dataset includes measurements obtained during two additional stress stimuli: the Stroop test and the Hyperventilation Provocation Test (HPT). Conversely, the WESAD dataset includes both the stress condition (TSST) and an amusement condition, which were alternated among different subjects to mitigate potential order effects. Consequently, in WESAD, there is one stress-anticipating interval for

Table 1. Comparison of WESAD and Stress-Predict Datasets

Dataset	Subjects	Duration	Labels (Duration)
WESAD (2018)	15	96 min (± 9 min)	Baseline (20 min), Amusement (6 min), Meditation (2×7 min), Stress (TSST: 10 min), Relax (total 47 min (± 9 min))
Stress-Predict (2022)	35	55 min (± 2 min)	Baseline (2×10 min), Relax (3×5 min), Stress (Stroop: 5 min, TSST: 10 min, HPT: 2 min)

each subject, whereas, in Stress-Predict, we consider all three stress-anticipating intervals stemming from the three stress stimuli.

The data from a single subject in the WESAD dataset is depicted in Fig. 1. Following each condition, subjects participated in a guided meditation to promote relaxation. During intervening periods, subjects were further provided opportunities to relax.

A comparison between the two datasets is provided in Table 1. In both study protocols, it was ensured that each stress condition was preceded and followed by a rest or meditation period lasting a minimum of five minutes. Baseline conditions aimed at inducing a neutral affective state were recorded at the beginning of each study, and at the end in the case of the Stress-Predict dataset. The datasets were labeled periodically, with specific time frames during the experiment categorized according to the perceived condition the subjects were placed under, as indicated in the Labels column of Table 1.

2.1 Predictive Modeling

As previously mentioned, we frame stress prediction as a four-class classification problem. Our methodology involves utilizing a time window of multiple physiological signals as input. These physiological signals encompass various parameters, including some derived from the recorded signals, as further detailed in this section. From each input window, a set of features is extracted to capture patterns in the data. These features are then utilized as input for a machine-learning classification algorithm, which assigns a class label. Similarly to other data stream handling methods, we utilize the sliding window technique to segment physiological signals into fixed-length intervals. We established the length of input windows to be 240 s. To mitigate potential information loss at window edges, consecutive windows are overlapping: the subsequent window is shifted by 30 s from the precedent.

Before feature extraction, input windows from each subject are separated into four classes: *no stress* (NS), *anticipating stress* (AS), *experiencing stress* (ES), and *post stress* (PS). These labels are assigned based on the stress sample proportions in the input window as well as its subsequent window. Taking into account the specific protocols for data collection in the analyzed datasets, which featured relatively short time spans between tasks, during which we hypothesize the subjects experienced stress anticipation, we chose for the subsequent window length to be equal to that of the input window. The labeling scheme is summarized in Table 2. Here, δ represents a threshold that determines the proportion of stress samples in the subsequent window below which the proportion is considered insignificant. We selected a δ value of 0.25 to balance the number of samples in the *anticipating stress* and *post stress* classes. Due to the selected input window and subsequent window lengths, along with parameters δ and shift, stress anticipation is observed within the seven-minute interval immediately preceding the stress condition. Figure 2 illustrates the application of the sliding window technique.

Table 2. Labeling approach: Input windows are labeled based on the proportion of stress samples within them and their subsequent windows.

Input window	Subsequent window	Label
0	$< \delta$	No stress
0	$\geq \delta$	Anticipating stress
> 0	$\geq 1 - \delta$	Experiencing stress
> 0	$< 1 - \delta$	Post stress

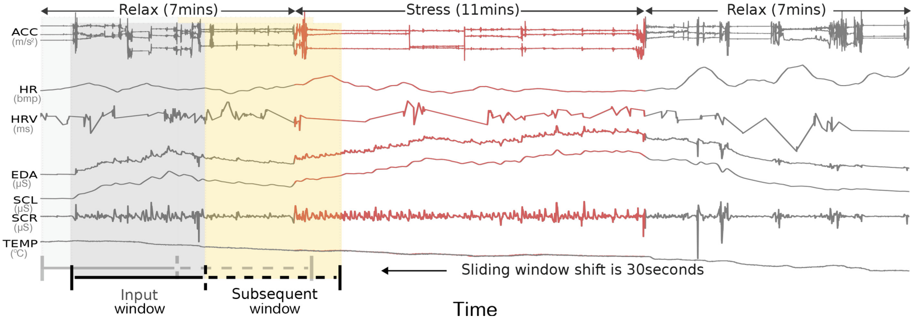


Fig. 2. Illustration of two sliding windows applied to a portion of Fig. 1-data: Each sliding window consists of an input window (gray area) and its subsequent window (yellow area). The consecutive sliding windows have a 30-second shift between them (Colour figure online).

Feature Extraction. The goal of feature extraction is to derive time-domain and frequency-domain features relevant to the classification task. We employ an approach aimed at capturing changes in feature values throughout the input window by segmenting it into overlapping windows. Specifically, we utilize a method that divides the input window into segments of 60 s each, with a 30-second overlap between consecutive segments, resulting in a total of seven subwindows per input window. Before feature extraction, we apply preprocessing techniques to the raw signals based on our review of relevant literature. Table 4 in the appendix presents the features extracted from different modalities. For the raw ACC signal, we compute various characteristics including mean, standard deviation, and peak frequency. These computations are performed separately for each axis as well as for the norm of the signal. We apply a pulse onset detection algorithm to process the raw BVP signal. After identifying pulse onsets, we compute the time intervals between consecutive pulses to construct a series representing the HRV signal. HR is derived by converting these intervals to beats per minute (bpm). Subsequently, features are derived from the windowed signals. Our approach to feature extraction from HR and HRV is aligned with that outlined in [20]. For the raw EDA signal, we initially applied a 5 Hz lowpass filter. We decompose the EDA signal into two components known as the skin conductance level (SCL) and skin conductance response (SCR) [11]. Regarding the raw TEMP signal, we compute common statistical features, including slope and dynamic range. In total, each one-minute subwindow produced 56 features, resulting in a representation of 392 features for one input window.

Classifiers. The previously extracted features are standardized to the same scale and form the entries of the input vector for the subsequent classification phase. We compare four machine learning algorithms: multinomial Logistic Regression (LR), Random Forest (RF), k-Nearest Neighbors (kNN), and AdaBoost (AB). These models were chosen for their popularity, compatibility with small datasets, and ability to balance accuracy with simplicity. Furthermore, they are suitable for practical deployment in devices with limited battery life, as demonstrated in previous studies [6,13,15]. To evaluate these algorithms, we employ repeated random sub-sampling validation, conducting ten repetitions for each subject's data. In our approach, a subject's dataset is randomly partitioned into three subsets of comparable size, maintaining the original class distribution. One subset is used for testing, while the other two are reserved for training. Before training the model, we address the class imbalance in the training set by employing random undersampling and the synthetic minority oversampling technique (SMOTE) [8] to ensure a consistent representation of 30 samples per class. The best hyperparameter combination is found through cross-validation. We fine-tuned parameters like k-nearest neighbors count, estimator numbers for Random forest and AdaBoost (capped at 150), and regularization for Logistic regression. After completing training and validation for each subject's dataset, we evaluate the model by testing it and by aggregating predictions across all subjects, and computing the performance metrics. Following training and testing for each subject's dataset, we aggregate predictions across all subjects and compute the performance metrics. We then assess each model's generalization and performance consistency across diverse subjects by aggregating results across the ten repetitions.

3 Experimental Results and Discussion

The classifiers' outcomes on the four-class classification task are shown in Table 3. Each machine learning method's performance is evaluated using precision and recall metrics for each class, in addition to average accuracy and F1-score. To assess these metrics for each class, we treat each class as a distinct binary classification task using the One-vs-Rest (OVR) strategy. The average F1-score is calculated as the unweighted mean of F1-scores across all classes, a technique known as macro-averaging [23].

Overall, the RF classifier achieves the highest prediction accuracy for both datasets, WESAD and Stress-Predict, with scores of 0.96 and 0.85, respectively. Similarly high accuracy values are attained by the LR model, scoring 0.94 and 0.79, respectively. Conversely, kNN and AdaBoost classifiers demonstrate notably inferior performance. Given the extensive feature set, it is likely that the data contains noise. This discrepancy in performance could be attributed to the generally lower robustness of kNN and AdaBoost classifiers to noise compared to RF and LR classifiers.

The RF classifier outperforms other classifiers in each class. However, the detection performance on the *anticipating stress* class is lower compared to classes *no stress*, *experiencing stress*, and *post-stress*. On the WESAD dataset, the RF classifier achieved a precision of 0.82 and a recall of 0.79, while on the Stress-Predict dataset, it attained a precision of 0.67 and a recall of 0.62. Moreover, RF achieves a much better performance in detecting *anticipating stress*, compared to the other models. This may be related to the specific capability of this model to prevent overfitting, which is a concrete risk in these small data sets. Moreover, it is worth noting a higher standard deviation across the ten iterations compared to other classes. A possible reason for this could be the increased variability among subjects in the data linked with the *anticipating stress* class, indicating that the model may need larger sample sizes to better capture underlying patterns.

To better illustrate the performance differences on the considered classes, Table 5 in the appendix presents the average confusion matrix generated by applying the RF classifier to the feature dataset related to a single subject. The corresponding visualization of recorded signals for this subject can be found in Fig. 1. The dataset for this subject comprises 195 data points, each characterized by 392 features, as detailed in Sect. 2.1. The test dataset is constructed with a stratified sampling approach, comprising two test samples for the AS class, and five each for the ES and PS classes while including 53 samples for the NS class. To ensure result stability and account for randomness, we conducted 100 repetitions of the experiment. Analysis revealed instances where AS observations were misclassified as NS or ES observations, which are adjacent classes to AS in terms of time. Similarly, PS observations were also misclassified as NS or ES observations. This suggests that the model may require larger sample sizes to better capture underlying patterns. Moreover, refining class boundaries through adjusting the parameter value δ, could improve the model's accuracy in capturing these patterns.

It is also worth noting that the WESAD dataset exhibits higher performance levels compared to the Stress-Predict dataset. This difference might stem from additional stress-inducing tasks in the Stress-Predict dataset, apart from the TSST. These additional tasks lack adequate evidence in the literature regarding their ability to reliably induce stress, unlike the TSST. Additionally, unlike the WESAD dataset, where substantial intervals between different tasks are present, the intervals between the three stress-inducing tasks are only five minutes in the Stress-Predict dataset, thereby constraining our analysis.

Table 3. Evaluation of classifiers on a four-class classification task using the WESAD and Stress-Predict datasets. Abbreviations: NS = no stress, AS = anticipating stress, ES = experiencing stress, PS = post-stress, LR = Logistic Regression, RF = Random Forest, *k*NN = *k*-Nearest Neighbors, AB = AdaBoost. Values represent mean performance metrics with standard deviations indicated by ± signs.

	WESAD									
	NS		AS		ES		PS		Accuracy	F1
	P	R	P	R	P	R	P	R		
LR	0.99 (± 0.01)	0.96(± 0.01)	0.61 (± 0.06)	0.70 (± 0.10)	0.84 (± 0.04)	0.91 (± 0.04)	0.81 (± 0.04)	0.83 (± 0.05)	0.94 (± 0.01)	0.84 (± 0.03)
RF	0.99 (± 0.0)	0.98 (± 0.01)	0.82 (± 0.08)	0.79 (± 0.09)	0.91 (± 0.04)	0.91 (± 0.05)	0.87 (± 0.04)	0.93 (± 0.03)	0.96 (± 0.01)	0.89 (± 0.03)
*k*NN	0.99 (± 0.0)	0.90 (± 0.01)	0.37 (± 0.05)	0.79 (± 0.08)	0.74 (± 0.03)	0.89 (± 0.04)	0.74 (± 0.03)	0.89 (± 0.04)	0.89 (± 0.01)	0.76 (± 0.02)
AB	0.96 (± 0.01)	0.97 (± 0.01)	0.58 (± 0.11)	0.48 (± 0.08)	0.80 (± 0.08)	0.80 (± 0.05)	0.82 (± 0.02)	0.80 (± 0.06)	0.92 (± 0.01)	0.77 (± 0.03)
	Stress-Predict									
LR	0.85 (± 0.04)	0.87 (± 0.04)	0.60 (± 0.05)	0.56 (± 0.06)	0.76 (± 0.04)	0.72 (± 0.04)	0.82 (± 0.03)	0.84 (± 0.03)	0.79 (± 0.02)	0.75 (± 0.02)
RF	0.90 (± 0.03)	0.92 (± 0.03)	0.67 (± 0.05)	0.62 (± 0.07)	0.84 (± 0.04)	0.78 (± 0.04)	0.86 (± 0.02)	0.89 (± 0.03)	0.85 (± 0.02)	0.81 (± 0.02)
*k*NN	0.86 (± 0.03)	0.86 (± 0.05)	0.51 (± 0.04)	0.72 (± 0.05)	0.78 (± 0.02)	0.77 (± 0.05)	0.87 (± 0.02)	0.77 (± 0.04)	0.79 (± 0.01)	0.76 (± 0.01)
AB	0.88 (± 0.04)	0.73 (± 0.05)	0.65 (± 0.08)	0.32 (± 0.05)	0.71 (± 0.04)	0.56 (± 0.04)	0.66 (± 0.02)	0.88 (± 0.02)	0.71 (± 0.02)	0.65 (± 0.02)

Principal Component Analysis. To gain insights into the patterns within the four classes, we conducted principal component analysis (PCA) on individual subjects' data. Below, we present an example using data from the subject analyzed earlier. Figure 3 displays the first two principal components and the top six loading vectors, while Table 6 in the appendix provides information about these loadings. The first loading vector places nearly equal importance on EDA-related measures but assigns less weight to HR, HRV, and TEMP-related measures. Conversely, the second loading vector prioritizes TEMP, HRV, and HR measures, assigning lower weight to the remaining features. Differences between classes for the subject can be explored through the two principal component score vectors depicted in Fig. 3 in the appendix. Observations with notably positive scores on the first principal component, such as those corresponding to *experiencing stress* and *post-stress*, exhibit high levels of skin conductance. Additionally, observations from these classes show negative scores on the second principal component, signifying lower values in skin temperature and heart rate variability. Data points near zero, primarily from the *no stress* class, on both components, suggest approximately average levels of skin conductance, heart rate variability, and skin temperature.

4 Conclusions and Future Work

This study analyzed biomarker data, including movement and physiological signals such as acceleration, electrodermal activity, heart rate, heart rate variability, and temperature, obtained from a wrist-worn device. Our objective was to evaluate the effectiveness of using these signals in detecting stress anticipation triggered by social evaluation. We used standard machine learning techniques and two publicly available datasets collected under laboratory conditions to explore this hypothesis, aiming to move beyond the typical binary stress vs. no stress classification task already considered in the literature. The experiments showcased the Random Forest's capability to accurately recognize the patterns of *anticipating stress* and distinguish them from other patterns, including *no stress*, *experiencing stress*, and *post stress*.

While this study provides valuable insights, it is important to note the limitation imposed by the relatively small sample size. Future research requires larger samples for validation, along with the development of specialized datasets tailored for studying stress anticipation. This study did not explore the impact of the combination of input window and subsequent window lengths, along with the choice of shift and parameter δ, on the system's prediction performance; this aspect will be investigated in future work.

This study primarily focused on prediction rather than exploring other aspects of the issue, such as strategies for effectively using these predictions. Questions persist regarding how to assist individuals in identifying optimal behaviors for change and providing ongoing support, crucial for closing the feedback loop with users. We aim for this study to emphasize the proactive potential of such systems.

Acknowledgments. We acknowledge the support provided by the PNRR initiatives: INEST (Interconnected North-East Innovation Ecosystem), project code ECS00000043, and FAIR (Future AI Research), project code PE00000013. These projects are part of the NRRP MUR program, funded by the NextGenerationEU. We extend our gratitude to the anonymous reviewers for their valuable feedback.

Appendix

Table 4. List of extracted features.

Signal	Features
ACC	mean, standard deviation, absolute integral, and peak frequency for each axis ACC_x, ACC_y, ACC_z, and the norm ACC_{norm}
BVP	HR and HRV: mean, standard deviation; HRV: number and percentage of intervals differing more than 50ms, energy in ultra-low (ULF), low (LF), high (HF) and ultra-high (UHF) frequency component; sum of frequency components in ULF-UHF range, root mean square of successive differences; ratio of LF and HF, normalized LF and HF component, relative power of each frequency component
EDA	EDA: mean, standard deviation, minimum and maximum values, slope, dynamic range; SCL and SCR: mean and standard deviation; SCL: correlation between SCL and time; SCR: number, sum, and total duration of identified SCR segments, area under the SCRs
TEMP	mean, standard deviation, minimum and maximum values, slope, dynamic range

Table 5. Average confusion matrix using Random Forest classifier for Fig. 1-data (averaged across 100 repetitions) with test sample counts.

	Predicted				Total samples
Actual	NS	AS	ES	PS	
NS	**51.29**	0.16	0.9	0.65	53
AS	0.07	**1.83**	0.1	0	2
ES	0.12	0.17	**4.27**	0.44	5
PS	0.03	0	0.13	**4.84**	5

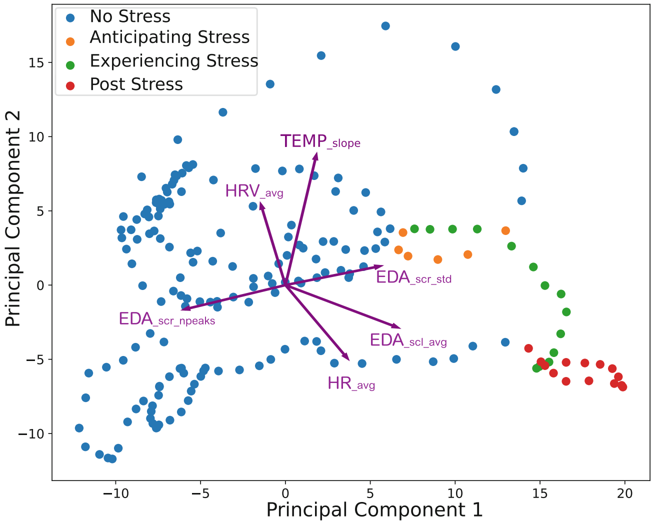


Fig. 3. Two principal components for the Fig. 1-data.

Table 6. Principal component loading vectors for the data in Fig. 1, scaled by a factor of 60 and shown in Fig. 3.

Feature	PCA1	PCA2
HRV_avg	−0.023	0.085
TEMP_slope	0.029	0.141
HR_avg	0.058	−0.078
EDA_scr_std	0.088	0.020
EDA_scr_npeaks	−0.094	−0.026
EDA_scl_avg	0.105	−0.045

References

1. Empatica E4 medical devices. http://www.empatica.com. Accessed 21 June 2024
2. Fitbit Sense 2 Smartwatch. https://bit.ly/3V5m7Z7. Accessed 21 June 2024
3. Allen, A.P., Kennedy, P.J., Dockray, S., Cryan, J.F., Dinan, T.G., Clarke, G.: The trier social stress test: principles and practice. Neurobiol. Stress **6**, 113–126 (2017)
4. Amanvermez, Y., et al.: Effects of self-guided stress management interventions in college students: a systematic review and meta-analysis. Internet Interv. **28**, 100503 (2022)
5. Amanvermez, Y.: Stress and stress management interventions in higher education students: Promises, Challenges, Innovations. Ph.D. thesis, VU E-Publishing (2023)
6. Andric, M., Ricci, F., Zini, F.: Sensor-based activity recognition and performance assessment in climbing: a review. IEEE Access **10**, 108583–108603 (2022)
7. Boucher, P., Plusquellec, P.: Acute stress assessment from excess cortisol secretion: Fundamentals and perspectives. Front. Endocrinol. **10**, 749 (Nov 2019)
8. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: Smote: synthetic minority over-sampling technique. J. Artif. Intell. Res. **16**, 321–357 (2002)
9. Dickerson, S.S., Kemeny, M.E.: Acute stressors and cortisol responses: a theoretical integration and synthesis of laboratory research. Psychol. Bull. **130**(3), 355–391 (2004)
10. Garmin: Garmin vívosmart® 4: Activity Tracker with Heart Rate Monitor & Fitness Tools, https://www.garmin.com/en-US/p/567813. Accessed 21 June 2024
11. Greco, A., Valenza, G., Lanata, A., Scilingo, E., Citi, L.: cvxEDA: a convex optimization approach to electrodermal activity processing. IEEE Trans. Biomed. Eng. **63**(4), 797–804 (2016)
12. Iqbal, T., et al.: Stress monitoring using wearable sensors: a pilot study and stress-predict dataset. Sensors **22**(21), 8135 (2022)
13. Ivanova, I., Andric, M., Janes, A., Ricci, F., Zini, F.: Climbing activity recognition and measurement with sensor data analysis. In: Companion Publication of the 2020 International Conference on Multimodal Interaction. ICMI '20, ACM
14. Jaimes, L.G., Gagneja, K., Akbas, M.I., Vergara-Laurens, I.J.: Future stress, forecasting physiological signals. In: 2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC), IEEE (Jan 2017)
15. Martinez, R., Salazar-Ramirez, A., Arruti, A., Irigoyen, E., Martin, J.I., Muguerza, J.: A self-paced relaxation response detection system based on galvanic skin response analysis. IEEE Access **7**, 43730–43741 (2019)
16. McEwen, B.S.: Protective and damaging effects of stress mediators. N. Engl. J. Med. **338**(3), 171–179 (1998)
17. O'Connor, D.B., Thayer, J.F., Vedhara, K.: Stress and health: a review of psychobiological processes. Annu. Rev. Psychol. **72**(1), 663–688 (2021)
18. Pan American Health Organization Caribbean Development Bank: Doing what matters in times of stress. Pan American Health Organization (Aug 2021)
19. Peake, J.M., Kerr, G., Sullivan, J.P.: A critical review of consumer wearables, mobile applications, and equipment for providing biofeedback, monitoring stress, and sleep in physically active populations. Front. Physiol. **9** (Jun 2018)
20. Schmidt, P., Reiss, A., Duerichen, R., Marberger, C., Van Laerhoven, K.: Introducing WESAD, a multimodal dataset for wearable stress and affect detection. In: Proceedings of the 20th ACM International Conference on Multimodal Interaction. ICMI '18, ACM (Oct 2018)

21. Schneiderman, N., Ironson, G., Siegel, S.D.: Stress and health: psychological, behavioral, and biological determinants. Annu. Rev. Clin. Psychol. **1**(1), 607–628 (2005)
22. Schulte-Frankenfeld, P.M., Trautwein, F.: App-based mindfulness meditation reduces perceived stress and improves self-regulation in working university students: A randomised controlled trial. Appl. Psychol. Health Well Being **14**(4), 1151–1171 (2021)
23. Sokolova, M., Lapalme, G.: A systematic analysis of performance measures for classification tasks. Inform. Process. Manage. **45**(4), 427–437 (2009)
24. Taylor, J.L., Muscatello, R.A., Corbett, B.A.: Differences in anticipatory versus reactive stress to social evaluative threat in adults versus adolescents with autism. Autism Res. **11**(9), 1276–1285 (2018)
25. Umberson, D., Liu, H., Reczek, C.: Stress and health behaviour over the life course. Adv. Life Course Res. **13**, 19–44 (2008)
26. Umematsu, T., Sano, A., Taylor, S., Tsujikawa, M., Picard, R.W.: Forecasting stress, mood, and health from daytime physiology in office workers and students. In: Proceedings of the 42nd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). IEEE (Jul 2020)
27. van der Valk, E.S., Savas, M., van Rossum, E.F.C.: Stress and obesity: are there more susceptible individuals? Curr. Obes. Rep. **7**(2), 193–203 (2018)
28. Vos, G., Trinh, K., Sarnyai, Z., Rahimi Azghadi, M.: Generalizable machine learning for stress monitoring from wearable devices: a systematic literature review. Int. J. Med. Inform. **173**, 105026 (2023)

Improving Reminder Apps for Home Voice Assistants

Abrar S. Alrumayh[1] and Chiu C. Tan[2](✉)

[1] Stevens Institute of Technology, Hoboken, NJ, USA
aalrumay@stevens.edu
[2] Temple University, Philadelphia, USA
cctan@temple.edu

Abstract. Voice assistants have emerged as a promising avenue for delivering healthcare services due to their accessibility and user-friendly nature. This paper explores the potential for enhancing these platforms through increased customization options for reminders. By leveraging the ambient sounds present in the home environment, such as household appliances or environmental noises, to provide additional context information to users. This approach aims to enrich the user experience, making reminders more informative and tailored to individual preferences and surroundings.

Keyword: Voice assistant · Smart health · Context awareness

1 Introduction

Home voice assistants (HVA), like the Amazon Alexa, Apple Homepod, and Baidu Xiaodu, are popular home devices that primarily relies on a *voice interface* for interaction. The user will interact with the device by talking using natural language, and the device will respond accordingly. HVAs are sometimes also called *Smart Speakers*, since their physical appearance resembles that of audio speakers. We use both terms interchangeably in this paper.

In terms of healthcare applications, smart speakers open up new possibilities for accessibility compared to traditional computers and touchscreen mobile devices [9].Research has demonstrated several potential benefits of using voice assistants in healthcare [8]. These technologies can enhance patient care and improve access to health information. At the same time, research has also indicated that current implementation of such systems in smart speakers may not be sufficiently effective or practical to fulfill user needs [7].

Our paper looks at the use of smart speakers to implement **reminders apps**. Reminder apps prompt or aid the user in remembering events, appointments, medication, etc., and play an important role in healthcare [2]. We present the design of reminder systems on smart speakers that make use of the sounds generated in the home to provide additional context information to decide whether to fire the reminder or adjust the reminder for later.

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 409–413, 2024.
https://doi.org/10.1007/978-3-031-66538-7_40

2 A Context-Aware Smart Speaker Reminder

Our experiences in testing our smartphone reminder app for older adults with cognitive problems [5] showed the importance of context-based reminders. HomeSmartPrompt uses the sounds in the home as cues to determine the context and decide whether to deliver the reminder, postpone the reminder, or prompt for context confirmation. The user will generate their list of reminders and their associated approximate times. Before the reminder goes off, it invokes the **Context Evaluator (CE)** that returns the user's current active context. The CE algorithm can return three possible outcomes (1) context is interruptible indicates the user is free, (2) context is un-interruptible indicates the user is busy and doesn't want to be distracted, and (3) uncertain.

If the current context is interruptible, the user is able to perform the task. As a result, the remainder will go off and waits for the response from the user. If no acknowledgment input from the users after some period of time, a follow-up reminder will go off prompt the users to confirm the completion of the task. The user also allows to snooze the reminder for any interval. If the current context is un-interruptible, that means the user cannot complete the task immediately. As a result, HomeSmartPrompt will wait for some period of time, then re-invoke the context evaluator to recheck if the context is changed. Otherwise, if the current context is Uncertain, that means the current context is ambiguous. As a result, HomeSmartPrompt will prompt for confirmation and use adaptive learning to learn from the user's response.

Figure 1 illustrates the process of the context evaluator. Given an audio recording, we segment an audio snippet *A*, and then remove the noise and filter out speech based off [6]. The audio input is converted to WAV format, resampled to 16 kHz mono, and filtered for frames below -16 dBfs amplitude. Mel-frequency cepstral coefficients (MFCC) are then extracted for later classification. We then attempt to determine the current active context within the home by comparing it against a set of pre-trained context models (Fig. 1 B). The model is pretrained on AudioSet from [4].

A home environment can have multiple kinds of sounds. We address this problem by splitting it into two components: (1) A group of single-sound binary classifiers (Fig. 1, C-1). These classifiers are trained individually for each ambient sound typically found within homes. (2) A context recognizer (Fig. 1, C-2). This component is trained on mixed ambient sounds using the probabilities outputted by each binary classifier. Its purpose is to determine the aggregate context by analyzing correlations between probability distributions of individual ambient labels. This separation facilitates adaptive learning for customization purposes. If the current context is ambiguous, it will announce the uncertainty, and start learning from the user's response. The adaptive model will assign the user's response to the current context and feed it along with the recorded audio to the adaptive engine dataset to train the model in the new context.

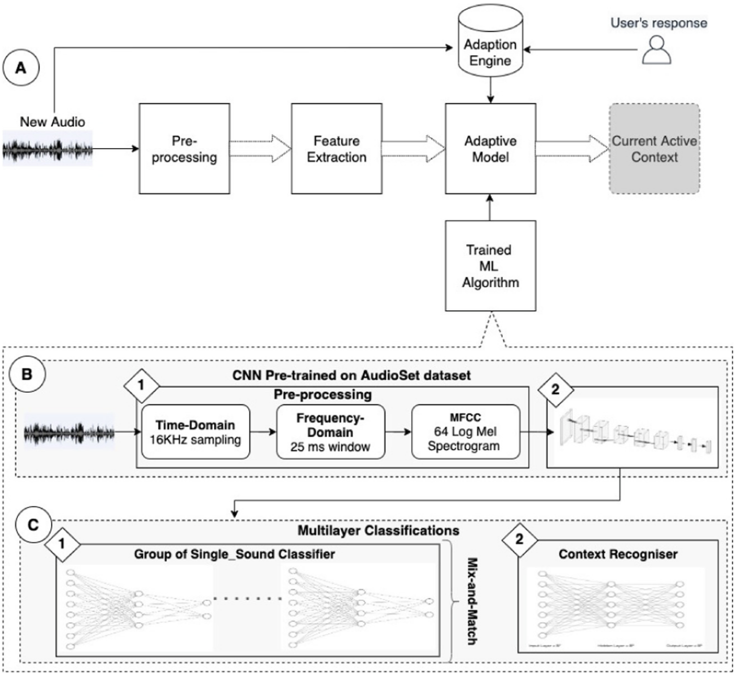


Fig. 1. Context evaluator training-flow overview

3 Evaluation

Since Amazon Alexa does not allow third-party application developers to access the audio before the wake word “Alexa” is uttered, HomePrompt cannot be built as an actual app. To test the feasibility of the context evaluator on a commercial platform, we emulated the working by using the Alexa drop-in function to replicate the environmental context sensing component. This is a feature that instantly connects a user with another Alexa device without requiring any action on the part of the other party. When a user drops in on an Alexa device, the device will chime once, and the light ring will pulse green, then the call will be answered automatically. As a result, the user will automatically hear anything within the range of the device. To capture audio recordings during drop-in interactions, we build a companion app to capture the audio when the trigger word is spoken; it saves three minutes during and after the trigger word is spoken. It then pushes a total of four minutes of audio to a secure server, we use Amazon Web Services AWS Lambda.

HomePrompt uses a pre-trained model to recognize context. For the evaluation, we considered the three different contexts: mealtime, having friends over, and user in the bathroom. These contexts offer realistic scenarios of the distraction situation and were selected based on [1] that conducted a user study to find personal contextual factors affecting interruptibility. For each context, we choose common sound events representing the context in home environments. In particular, nine sound events that consist of conversation, plate clattering, music, laughing, crowd, television, water running, toilet flushing, and tooth brushing. These sounds are chosen because they are the common distinctive acoustic characteristics that happen frequently in these contexts. We used the sounds from the publicly available library FreeSound [3] to create the pre-trained model.

For the evaluation environment, we conducted the experiments by placing the audio source next to the Echo device. We split the audio dataset into the train and test set with a ratio of 70% and 30%. We played the audio files using a laptop MacBook Pro to send as input to Alexa. The volume levels of the audio source set to a medium of about 73 dB were averaged across two listeners and measured in decibels (dB) using the Decibel X app[1].

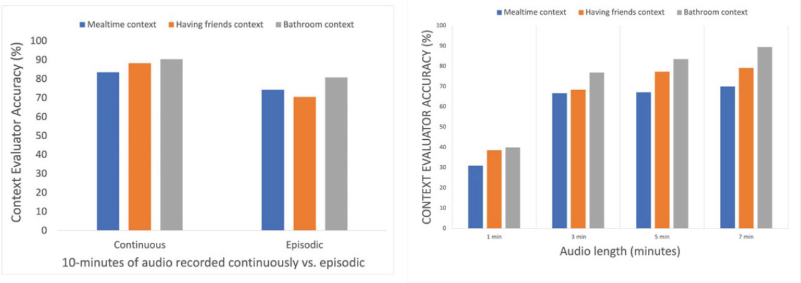


Fig. 2. (a) Accuracy of recognition. (b) Accuracy as a function of recording length.

Comparing Continuous and Episodic Listing Strategies. First, we explore the effect of listing strategies by comparing two sampling methods: **Continuous:** For an alarm that is set at time X, we will start listing at time Y for Z period of time. **Episodic:** For an alarm set at time X, we will start sampling for the K period of time before J time. To explore the effect of listening strategy on the accuracy of context recognition, we compare the accuracy of the CE algorithm from long continuous recording versus aggregation of the shorter recording snippets. A five-minute clip and fifteen 10-s snippets are extracted randomly from the audio recording. Figure 2(a) shows that the two tests give relatively close outcome, with an average accuracy of 87.4% and 78.1% from the 10-minute clip and across all snippets, respectively. This represents only a 9.3% increase in the accuracy of continuous recording over the aggregated snippets across all contexts. This experiment reveals the fact that using short snippets of audio does not necessarily affect recognition accuracy.

Then, we evaluate the accuracy of the Context Evaluator CE Algorithm with respect to the recording length. Figure 2 (b) shows the average accuracy for different audio lengths: 1, 3, 5, and 7. We found that with audio of length 2 minutes, we have the lowest accuracy for all test sets. On the other hand, audio of length 7 has the highest accuracy. We also find that the accuracy decreased by at most 3% with the audio of length 5, which guarantees that a good performance is achievable even when the length of the audio is reduced. Therefore, we believe that 5 min segments represent a good balance between processing time and accuracy while upholding user privacy.

[1] https://apps.apple.com/us/app/decibel-x-db-sound-level-meter/id448155923.

4 Conclusions

Smart speakers, such as Google Home or Amazon Alex, provide aid services that help users to create a reminder of tasks that they want to accomplish that they may otherwise forget. We designed and built a voice-activated reminder app on Amazon Alexa incorporating home-generated sounds to provide additional context information.

References

1. Cha, N., et al.: Hello there! Is now a good time to talk? Opportune moments for proactive interactions with smart speakers. In: Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, vol. 4, no. 3, pp. 1–28 (2020)
2. Fenerty, S.D., West, C., Davis, S.A., Kaplan, S.G., Feldman, S.R.: The effect of reminder systems on patients' adherence to treatment. Patient Prefer. Adherence, 127–135 (2012)
3. Font, F., Roma, G., Serra, X.: Freesound technical demo. In: Proceedings of the 21st ACM International Conference on Multimedia, pp. 411–412 (2013)
4. Gemmeke, J.F., et al.: Audio set: an ontology and human-labeled dataset for audio events. In: 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 776–780. IEEE (2017)
5. Hackett, K., et al.: Remind me to remember: a pilot study of a novel smartphone reminder application for older adults with dementia and mild cognitive impairment. Neuropsychol. Rehabil., 1–29 (2020)
6. Hershey, S., et al.: CNN architectures for large-scale audio classification. In: 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 131–135. IEEE (2017)
7. Kim, S., et al.: Exploring how older adults use a smart speaker–based voice assistant in their first interactions: qualitative study. JMIR Mhealth Uhealth **9**(1), e20427 (2021)
8. Kocaballi, A.B., et al.: Conversational agents for health and wellbeing. In: Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems, pp. 1–8 (2020)
9. Sunshine, J., et al.: Smart speakers: the next frontier in mhealth. JMIR Mhealth Uhealth **10**(2), e28686 (2022)

Author Index

J. Finkelstein et al. (Eds.): AIME 2024, LNAI 14844, pp. 415–418, 2024.
https://doi.org/10.1007/978-3-031-66538-7