Received 13 November 2025, accepted 2 December 2025, date of publication 5 December 2025,
date of current version 17 December 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3641034

RESEARCH ARTICLE

A Domain-Agnostic Neuro-Symbolic Architecture for Multimodal Human-in-the-Loop Anomaly Detection and Complex Fault Diagnosis

TIM BOHNE[1,2], ANNE-KATHRIN PATRICIA WINDLER[2], AND MARTIN ATZMUELLER[1,2]
[1]Semantic Information Systems Group, Osnabrück University, 49090 Osnabrück, Germany
[2]German Research Center for Artificial Intelligence (DFKI), 49084 Osnabrück, Germany

Corresponding author: Tim Bohne (tbohne@uni-osnabrueck.de)

The DFKI Niedersachsen is sponsored by the Ministry of Science and Culture of Lower Saxony and the Volkswagenstiftung. The paper describes work carried out in the context of the funded projects AW4.0 and Railway-X (Bundesministerium für Wirtschaft und Klimaschutz (BMWK), Grants: 68GX21005D, 13IPC039B).

ABSTRACT This paper presents a general architecture for iterative, hybrid neuro-symbolic anomaly detection and complex fault diagnosis, in which symbolic knowledge-based methods and neural machine learning methods reinforce each other. For evaluation, we introduce a neuro-symbolic diagnosis benchmark that systematically assesses the architecture using randomized, parametrized synthetic problem instances with ground truth solutions. These are derived from an abstract formalization of the general problem of diagnosing systems composed of causally interconnected components based on sensor signal evaluation. It results in a domain-agnostic diagnostic framework, where synthetic instances capture a multitude of practical domains, enabling robust, empirically grounded conclusions. Explainability and interpretability emerge naturally through the specific neural-symbolic interplay. The architecture serves as a transferable blueprint for diagnosing systems across domains involving causal structure and sensory assessment.

INDEX TERMS Neuro-symbolic AI, knowledge representation, anomaly detection, fault diagnosis, explainable AI, interpretability.

I. INTRODUCTION

The automated diagnosis of complex systems, e.g., technical machinery, is a challenging task. From its inception, it has been a prominent area of research [1], [2], [3], for instance, using ontology-based approaches [4], [5], [6], [7], [8], [9], [10], often complemented by learning methods [11], [12], [13], [14], [15], [16]. To successfully implement either approach, adequate knowledge or data is required. However, the interconnection between the systems, e.g., for handling different types of information in respective abstractions, is not inherently provided. Furthermore, knowledge acquisition is often costly, while purely data-driven techniques require large amounts of data of sufficient quality. We study knowledge- and machine-learning-based fault diagnosis, combining both paradigms. The approach involves an iterative diagnosis cycle in which preliminary hypotheses are refined using both knowledge-based and data-driven methods. Explainability and interpretability are crucial for diagnosis and emerge naturally through the specific neural-symbolic interplay. In order to be able to interpret and judge the results of anomaly detection, methods of *eXplainable Artificial Intelligence* (XAI) [17], [18], [19] are employed so that not only accurate predictions are obtained but, moreover, predictions that are comprehensible for humans. Accordingly, the objective is not to replace human experts, but rather to support them. Specifically, the aim is to provide a general diagnosis framework that may be instantiated in various domains, e.g., in the one from [20]. This paper is an adapted and significantly expanded version of our previous work [20]. In particular, we provide a domain-agnostic generalization, extension, and, most importantly, systematic evaluation of our approach presented in [20]. The evaluation goes way beyond a particular domain and provides a formal specification and analysis of general variable properties of conceivable diagnostic domains involving causal structure and sensory

The associate editor coordinating the review of this manuscript and approving it for publication was Jiawei Yang.

assessment. The effectiveness and feasibility of the approach in a real-world scenario has been demonstrated in [20]. This paper complements it with a thorough theoretical analysis and quantitative evaluation. A neuro-symbolic approach to the problem of automated or semi-automated diagnosis has a major advantage: Compared to previous methods relying solely on knowledge-based or data-based techniques, the neuro-symbolic architecture is designed in such a way that both paradigms are mutually beneficial. Thus, it is motivated by the previous lack of interpretability and exploitation of available domain knowledge in data-driven methods, and the extensive manual effort and shortcomings with respect to sensor signal evaluation in expert systems. Our core contributions are summarized as follows:

1) *Generalization and extension of the neuro-symbolic diagnosis architecture presented in [20], featuring explainability, interpretability, and knowledge discovery.*
2) *Abstract formalization and analysis of the general problem of diagnosing systems with causally interconnected components based on sensor signal evaluation.*
3) *Neuro-symbolic diagnosis benchmark featuring a systematic evaluation of the architecture using randomized, parametrized synthetic problem instances and corresponding ground truth solutions generated based on the established formalism.*

The subsequent sections of the paper are organized as follows: Section II discusses related work. After that, Section III presents the method and architecture, before further elaborating the symbolic knowledge representation in Section IV. Subsequently, Section V describes the ANN-based signal classification and Section VI the actual diagnostic process. Section VII not only provides a systematic evaluation, but also a thorough formalization of the underlying theoretic family of diagnosis problems, as well as reflections on relationships between certain aspects of the problem structure. Finally, Section VIII concludes with a summary and promising directions for future research.

II. RELATED WORK

In the following, we discuss related work on (1) knowledge-based, (2) neural network-based, and (3) neuro-symbolic methods, as well as important distinctive features of our proposed approach and architecture.

Knowledge-Based Methods. There have been various attempts at knowledge-based diagnosis, primarily from the field of expert systems [4], [5], [9], [21], [22], [23]. Furthermore, the diagnosis of complex systems that generate large amounts of data, for instance, in industrial contexts [9], [24], [25], benefits from structured data capture in an ontology [26], e.g., via on-board sensors. This can reduce the need for extensive manual data preparation by experts, which is the focus of the latter work, while not providing a diagnostic method. In contrast, [27] defines an automotive ontology to capture the dependencies between different vehicle components and subsystems, emphasizing the modeling of fault propagation between components. This is used to substitute missing signals in monitored sensor recordings, while a *diagnostic directed acyclic graph* guides the stepwise, online fault diagnosis process. Yet, the work does not explain in detail how the sensor recordings are evaluated. In general, knowledge-based methods are not only relevant for technical systems [28], [29], [30], [31], but also for biological systems [32], [33], [34] or approaches in the medical domain [2], [35], [36]. Particularly for such high-risk domains, explainable approaches supported by large amounts of available data in conjunction with existing domain knowledge are crucial. This is the focus of the present work, in which we introduce such a comprehensive approach along with an in-depth, domain-agnostic evaluation.

Neural Network-Based Methods. Convolutional Neural Networks (CNNs) have shown to be highly effective in both image and time series classification [37], [38]. Anomaly detection [39], [40], [41], which can be seen as a binary classification task, has also been effectively addressed in practical applications using CNNs, among other neural architectures [42], [43], [44]. For instance, [45] applies a CNN to detect magnetic anomalies. Additionally, CNNs offer the advantage of explainability, e. g., through *Class Activation Mapping* (CAM) techniques [46], [47], providing explanatory insights into the temporal / spatial segments that are important for the network's prediction. Anomalies in the context of this paper refer to specific, known fault cases. Therefore, it is a matter of recognizing specific faults and not simply deviations from the norm. It is thus binary in the sense that either a specific fault is detected or the signal is classified as regular. Also, the uncertainty allows further conclusions on whether an anomaly is known (high confidence) or unknown (high uncertainty). Moreover, anomalies in this work can generally be divided into three categories in accordance with the definition in [48]: *subsequence anomaly in univariate time series*, *contextual anomaly in univariate time series* and *contextual anomaly in multivariate time series*. Our proposed approach can be considered explainable with regard to those anomalies, supporting a range of explanation methods.

Neuro-Symbolic Approaches. Recently, hybrid or neuro-symbolic approaches for fault diagnosis have emerged – featuring the reasoning capabilities of symbolic knowledge [49], [50], [51]. These allow the integration of domain-specific information and relationships while simultaneously utilizing the pattern recognition capabilities of, e. g., CNNs. [51] proposes such a hybrid model, using a CNN-based architecture for fault classification of sensor recordings. The classification result is forwarded to a knowledge graph (KG) [52], [53], which retrieves further details such as fault location, maintenance method, etc., based on historical data. [54] uses CNNs to classify different types of faults in power grids based on voltage waveforms. They combine it with *Answer Set Programming* [55], a logic-based module that is used to make a final prediction in case of uncertainty of the CNN. Symbolic rules encode the domain knowledge used to infer the class. In both hybrid approaches, the primary task is CNN-based classification, while the symbolic component assists by providing additional knowledge; however, without including any form of explainability. [56] proposes a method for knowledge-informed fault diagnosis in the area of bearing fault detection, utilizing interpretable signal processing with statistical and logical operators. Yet, the authors do not provide a general method combining ontologies with deep learning as proposed in this paper.

Discussion. In our previous work [20], we proposed a neuro-symbolic procedure that combines KG-based diagnosis with CNN-based anomaly detection, similar to [51], but with the main distinction that the primary task is an iterative diagnosis process guided by KG-based symbolic reasoning, while CNN-based anomaly detection is used as an assisting subroutine. Hence, according to the taxonomy in [57] that categorizes neuro-symbolic systems based on the integration of the neural and symbolic components, the model belongs to the *Symbolic[Neuro]* category. The KG initiates the anomaly classification networks as needed and uses the classification result to guide the subsequent diagnostic step. Explainability and interpretability are core features of the proposed procedure. To illustrate, XAI methods are used to generate visual explanations for classifications, always embedded in the overall diagnostic process. Ultimately, a comprehensive explanation is constructed by providing all of these diagnostic artifacts as an explanatory report. To the best of the authors' knowledge, this was the first fault diagnosis system in the *Symbolic[Neuro]* category.

This paper generalizes the approach presented in [20] and provides a framework for multimodal, human-in-the-loop anomaly detection and complex fault diagnosis. We consider the general problem of diagnosing systems composed of causally interconnected components, each associated with some form of sensory input. Additionally, it formalizes the abstract problem and systematically evaluates the framework using synthetic instances to identify limitations and gain insights implicitly covering many practical domains, e. g., the one in [20]. We consider the integration of machine learning and XAI into knowledge-supported fault diagnosis systems. Consequently, we present a novel, generalized, domain-agnostic, neuro-symbolic diagnosis system whose concrete instantiation in a practically relevant real-world context, specifically the automotive domain, has previously been demonstrated. Furthermore, we thoroughly evaluate it based on a systematic analysis and formalization of the general underlying diagnosis problem to allow an understanding of its potential application areas as well as its limitations.

III. METHOD AND ARCHITECTURE

The overall notion is to have a *knowledge graph* (KG) that guides the diagnostic process, coupled with *neural networks* that enable the interpretation of sensor signals suggested by the KG for investigation. Accordingly, the aim is to extract meaning from multimodal data that is unmanageable for humans and to recognize complex patterns. The general architecture of an according hybrid neuro-symbolic diagnosis framework is shown in Fig. 1. The diagnostic reasoning, i. e., the meaningful connection of the neural and symbolic parts, is represented by the *diagnostic circuit*, which is initialized with a domain-specific *fault context*. Based on the state, the KG is queried for expert knowledge required for diagnosis, e. g., aspects to measure based on the provided context. All these requests are managed by the *knowledge graph query tool*, which translates them into SPARQL queries to the KG and processes the responses.

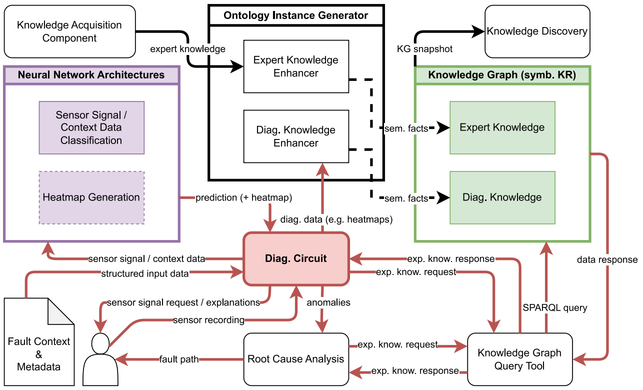


FIGURE 1. Neuro-symbolic architecture [20].

The typical case is that the KG suggests a component or, generally, a measurable property of the diagnosed system to be captured by some sensor. This sensor signal is the input to the neural side of the framework, which performs *binary classification (anomaly detection)* and, if applicable, *heatmap generation* to provide a visual explanation of the prediction. If one or more anomalies are detected in this way, the problem is isolated by performing a *root cause analysis* (RCA), which, in turn, relies heavily on the KG – containing causal relations between components of the diagnosed system. Whenever an anomaly is detected, all other components of the system whose malfunction could affect the correct functioning of the currently considered component are recursively investigated. Eventually, this leads to a *fault path* that starts at the probable root cause of the problem and cascades to other components of the system. As indicated by this example, diagnosis is expected to take place at the abstraction level of the components of the diagnosed system. According to Fig. 1, there are two types of knowledge modeled in the KG: (1) the *expert knowledge* structurally acquired via a *knowledge acquisition component*; (2) *diagnostic knowledge*, for which every relevant piece of diagnostic data (sensor recordings, predictions, heatmaps, etc.) is entered into the KG during the diagnostic process for exploitation in *knowledge discovery*. The semantic facts for both types of knowledge are created by an *ontology instance generator*.

In summary, the knowledge-based part of the framework establishes an initial hypothesis that is iteratively refined throughout the diagnostic process using artificial neural networks (ANNs). It starts with a component suggested by

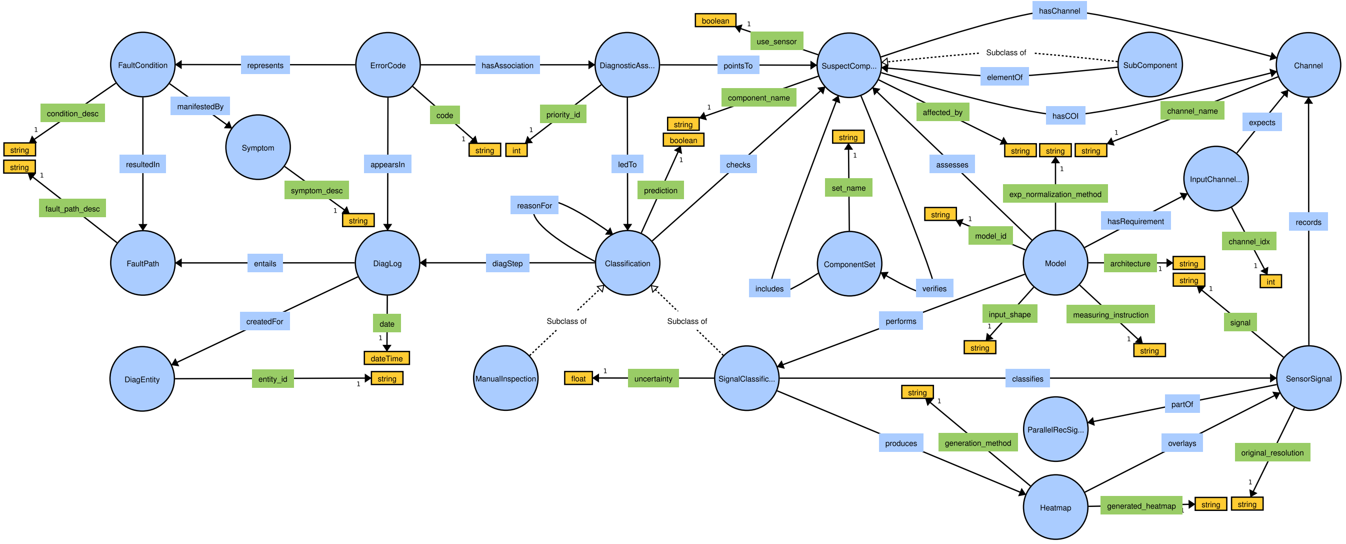


FIGURE 2. Abstract ontology for capturing diagnostics knowledge (generalizing and extending [20]).

the KG based on a fault context, but not necessarily the root cause of the problem. Therefore, it proceeds with fault isolation (RCA). Once the problem is isolated, the result is a multi-element fault path starting at the probable root cause.

IV. SYMBOLIC KNOWLEDGE REPRESENTATION

To capture and structure diagnosis-relevant knowledge, an ontology[1] was defined (cf. Fig. 2, generalizing and significantly extending the ontology from [20]), on the basis of which a KG emerges by populating it with large amounts of instance data. Essentially, there are three levels of abstraction: The raw definition of the ontology, entity-agnostic *expert knowledge* (cf. Sec. IV-A), and entity-specific *diagnostic knowledge* automatically acquired as part of the diagnostic process (recorded sensor data, explanations, etc., cf. Sec. IV-B). However, as illustrated in Fig. 1, the two types of knowledge are not isolated from each other, but connected by meaningful links (e. g., connecting classification instances to the diagnostic associations that led to them) to learn from previous diagnostic runs.

All three levels combined constitute the KG. The acquisition of expert knowledge is accomplished via a web interface (collaborative *knowledge acquisition component*) through which the knowledge is entered, stored in the *Resource Description Framework* (RDF) format, and hosted on an *Apache Jena Fuseki*[2] server. In addition, libraries have been developed that render this knowledge retrievable in the diagnostic process via predefined SPARQL queries (*KG query tool*, cf. Sec. IV-E), as well as making the KG expandable and editable in general (*ontology instance generator*, cf. Sec. IV-C, IV-D). To establish a connection to the KG hosted by the *Fuseki* server, i. e., to perform queries as well as KG extension via HTTP requests, a connection controller has been implemented. It sends HTTP requests containing the respective encoded SPARQL queries to the `/sparql` endpoint of the KG server. Furthermore, it sends HTTP requests with the serialized semantic facts (RDF triples) to be entered into the KG to `/data`, as well as HTTP fact deletion requests (`DELETE DATA` queries) to `/update` in order to remove deprecated knowledge.

A. EXPERT KNOWLEDGE MODELED IN THE ONTOLOGY

At the core of the knowledge captured in the ontology are the error codes, which are a perfect example of what is meant by *fault context* in the neuro-symbolic architecture visualization in Fig. 1. An `ErrorCode` can have `DiagnosticAssociations` with physical components that are part of the entity of diagnosis (`SuspectComponent`). A crucial aspect of such an association is the `priority_id`, based on which components are suggested to be examined in a certain order in the presence of a given error code. In addition, `affected_by` represents a list of other components whose malfunction could affect the correct functionality of the considered component (dependencies can be conceived as a tree, cf. Fig. 9). Domain experts can define a `ComponentSet` to reduce the number of redundant diagnostic steps in case there is a specific component that can be leveraged to verify the correct functioning of a whole set of components. Moreover, each error code represents a `FaultCondition`. Obviously, due to the tremendous variety of diagnostic entities in most domains, e. g., vehicle models in [20], and the constant development of additional ones, expert knowledge will usually never be exhaustive. Since the diagnosis is component-based, it is feasible to progress from use case to use case so that the system supports an increasing number of subsystems over time. Consequently,

[1] https://github.com/tbohne/nesy_diag_ontology/releases/tag/v0.0.3

[2] https://jena.apache.org/documentation/fuseki2/

it is crucial to have an efficient and structured knowledge acquisition, which is covered in section IV-C. The matter of correctness is assumed to be handled via crowdsourcing, with the pool of associated experts having the ability to refine entries.

B. DIAG. KNOWLEDGE MODELED IN THE ONTOLOGY

As anticipated, there is another theme to the ontology, which is the acquisition and reasonable arrangement of diagnostic data. For each `DiagEntity` instance that is entered into the KG, i. e., for each entity that is diagnosed with the system, a `DiagLog` is created that provides the KG with a kind of explanatory summary of the entire diagnostic process. However, this is not a mere summary, but each entry is automatically sorted into the existing web of expert knowledge and past diagnostic data by instantiating the concepts of the ontology. Initially, any recorded error code appears in this log, as this is always the starting point for a diagnosis. Perhaps most significant are the diagnostic steps, which are also part of the log in the form of `Classification` instances that store their reason, either another classification that detected an anomaly (`reasonFor`) or a diagnostic association with an error code recorded in the entity (`ledTo`). Each classification has a binary result (`prediction`) that indicates whether the checked component has an anomaly or not. The concept `Classification` has two sub-concepts: `ManualInspection` is a classification performed manually by a human. This is necessary in cases where sensor signal-based analysis is infeasible for a component, i.e., `use_sensor` := `false`. The other sub-concept is `SignalClassification`, which classifies a signal using a classification model (cf. Sec. V). In this case, we specify an uncertainty value, an ID and various preprocessing and architectural information of the model that produced the classification. The target of a signal classification are the `SensorSignals`, which are also stored in the KG and possibly grouped as an instance of a `ParallelRecSignalSet`. The ontology in Fig. 2 not only generalizes the concepts of the ontology presented in [20], but also extends them considerably, e.g., by including multivariate signals and the corresponding concepts: `SubComponent`, `Channel`, `InputChannelRequirement`, etc. The idea is that a signal can consist of more than one channel and a corresponding model expects the multivariate, i.e., multichannel signal in a specific order. Finally, we also provide heatmaps for the classification of the signals, which allow an interpretation of the predictions (cf. Sec. V-A). `Heatmaps` are stored in the KG along with their generation method. The final diagnosis takes the form of a series of `FaultPaths` that start with one component (root cause) and then cascade to others. Fault paths are not only stored in the diagnostic log, but also associated with fault conditions for knowledge discovery potential.

C. ENHANCEMENT OF EXPERT KNOWLEDGE

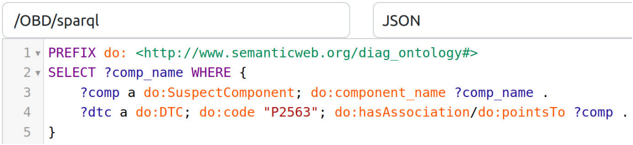
The *expert knowledge enhancer* can be used to augment the KG hosted by the *Fuseki* server with entity-agnostic expert knowledge. In particular, it generates semantic facts based on the information entered through a web interface and connects these facts in a meaningful way to what is already available in the KG, i. e., it serves as a backend for the *knowledge acquisition component*. Alternatively, it is possible to select an existing instance, view the currently available data, and refine it. Entering a new instance leads to a series of operations in the backend. Thus, it is always expert knowledge input via the web interface and corresponding generation of semantic facts in the backend. All of this is accompanied by a series of input validation mechanisms. This way, a simple KG extension for the expert goes hand in hand with an automatic proper "wiring" of semantic facts in the background.

D. ENHANCEMENT OF ENTITY-SPECIFIC DIAGNOSIS KNOWLEDGE

The *diag. knowledge enhancer*, on the other hand, enhances the KG with diagnosis-specific instance data, i. e., it connects the data recorded in a particular entity, as well as sensor readings, classifications, etc. generated during the diagnostic process, with corresponding background knowledge stored in the KG. The process typically starts by creating an instance of the entity to be diagnosed. Likewise, it typically ends with a call to `extend_kg_with_diag_log`, which takes numerous arguments, including the error code instances that are part of the diagnosis and the entity ID. This leads, for instance, to `FaultCondition` "*Boost Control Position Sensor Circuit: Implausible Signal*", represented by `ErrorCode "P2563"`, `resultedIn` a `FaultPath`, entailed by a `DiagLog` instance, `createdFor` the `DiagEntity` instance with `entity_id "ID2342713"`. Moreover, `"P2563" appearsIn` this particular instance of `DiagLog`. These are only a few examples of the information collected during diagnosis and its interrelationships. The full set of concepts and relations can be seen in the visualization of the ontology (cf. Fig. 2), for all of which automatic semantic fact generation and thus KG extension is implemented.

E. KNOWLEDGE GRAPH QUERY TOOL

Eventually, there is a library of numerous predefined SPARQL queries and response processing to access information stored in the KG that is used in the diagnostic process, such as `query_entity_instance_by_id(id)` and `query_suspect_components_by_error_code (code)`. The latter, for instance, automatically sends and processes the query shown in Fig. 3 for a given error code `"P2563"`. Note that `DTC` is the particular type of error code used in [20] and can be seen as an instantiation of `ErrorCode`.



/OBD/sparql

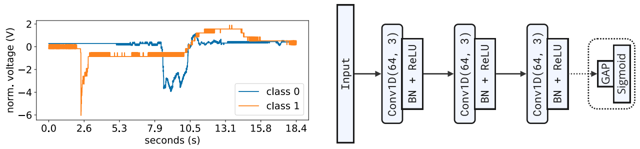
JSON

```
PREFIX do: <http://www.semanticweb.org/diag_ontology#>
SELECT ?comp_name WHERE {
    ?comp a do:SuspectComponent; do:component_name ?comp_name .
    ?dtc a do:DTC; do:code "P2563"; do:hasAssociation/do:pointsTo ?comp .
}
```

FIGURE 3. SPARQL query to retrieve components [20].

V. ANN-BASED SIGNAL CLASSIFICATION

A key idea of the developed diagnosis system revolves around sensor information. Signal recordings are performed on specific physical components in the entity of diagnosis to detect indications of problems (anomalies). The recorded signals are fed into a classification model previously trained on a large dataset, which evaluates whether each recording contains anomalies.[3] In this case, the task comes down to binary univariate / multivariate time series classification. In general, however, the architecture does not assume the signals to be time series; it would also work with images, for instance. To give an example: In [20], the battery voltage during the engine starting process is used. For battery voltage records $V \in \mathbb{R}^n$, performance was best when standard z-normalization was applied to the raw time series data, i.e., $V' := \{\frac{x_i - \mu V}{\sigma V} \mid x_i \in V\}$, with mean μV and standard deviation σV. Fig. 4a shows a regular (1) and an anomalous (0) z-normalized voltage sample. Since the main focus of this work and [20] is not to propose a superior ANN architecture for binary classification (anomaly detection) of time series data, we compared several standard architectures from the literature, made slight adjustments, and selected the best performing one for our purposes. In the case of the univariate signals under consideration, this was a *Fully Convolutional Network* (FCN). The FCN model (shown in Fig. 4b) is based on [58], in which the authors propose a strong baseline architecture for time series classification. Details concerning the training setup, data, etc. can be read in [20]. For multivariate signals, we trained and applied XCM (*Explainable Convolutional Neural Network for Multivariate Time Series Classification*) [59] models.



(a) Norm. Battery Voltage [20] (b) FCN Architecture [20]

FIGURE 4. Battery voltage signal classification.

A. SALIENCY MAP GENERATION FOR TIME SERIES

Explainability and interpretability should be core features of diagnosis systems. They arise relatively naturally in the system proposed in [20] and generalized in this work through the specific interplay of neural and symbolic methods. XAI methods are used to generate visual explanations for time series classifications as an exemplary type of artifact, always embedded in the overall knowledge-based diagnostic process. Ultimately, a comprehensive explanation is constructed by contextualizing all these diagnostic artifacts with symbolic state transitions as an explanatory report. Additionally, they augment the KG and enable to learn the most significant aspects of the signal types over time. Despite the commonplace of a black box nature of deep learning approaches, CNNs, among other architectures, offer the advantage of explainability, e. g., through *Class Activation Mapping* techniques [46], [47], providing explanatory insights into the temporal / spatial segments that are important for the network's prediction. Thus, subsequent to the classification of a signal, the explanation of the decision proceeds on the basis of *Class Activation Maps* (CAMs). This is to ensure that humans do not have to "blindly" rely on the model's predictions, which should reduce the proneness to errors, and can further enable computational sensemaking towards supporting humans [60], [61]. There are several techniques used in deep learning to visualize areas of an image that are most relevant to predicting a certain class, e. g., *Grad-CAM* [46], *HiResCAM* [62], *Grad-CAM++* [63], *Score-CAM* [64], *SmoothGrad* [65], and *LayerCAM* [66]. They provide a way to interpret the decision made by an ANN model (with compatible architecture, unless model-agnostic) by highlighting the regions of the input image that contribute the most to the classification result. The details of its implementation and how the methods are applied to time series data can be read in [20]. Eventually, saliency maps can be generated for a time series classification model analogous to the standard case with image data. Each of the methods receives the normalized time series values $V' \in \mathbb{R}^n$, the trained model M, and an optional prediction y (default is the "best guess", i. e., $y = \text{argmax}_i\, P(i \mid V')\, \forall\, i \in \{0, 1\}$) as input, and outputs a heatmap $H \in [0, 1]^n$ highlighting the parts that are most relevant for the classification. We are interested in values $h \in H$ close to 1, these are the most important parts of the signal V'. Each value $h_i \in H$ rates the importance of a corresponding input value $v_i \in V', \forall\, i \in \{1, \dots n\}$. Since the different methods have different advantages and weaknesses, the best-suited depends on the considered task. Fig. 7 shows a side-by-side plot of all generated heatmaps. As can be observed in the visualization, in this case the different methods agree very well on the relevant regions for the prediction. In the end, it enables domain experts to assess whether these areas are plausible bases for decision making and allows for knowledge discovery through the resulting KG entries. In the case of multivariate signals, a heatmap is created for each channel, i.e., for each synchronously considered variable over time. Fig. 5 shows the *variable attribution maps* for four parallel recorded signals generated with the *Grad-CAM* method. These highlight the segments that were most relevant for the prediction when extracting information from each variable separately. Additionally, since anomalies can arise from the specific interplay of

[3] https://github.com/tbohne/oscillogram_classification/releases/tag/v0.2.0

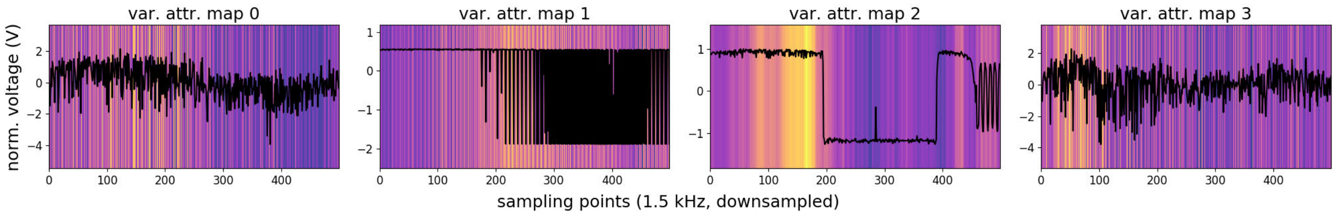


FIGURE 5. Variable attribution maps of four synchronously recorded signals using real-world automotive sensors.

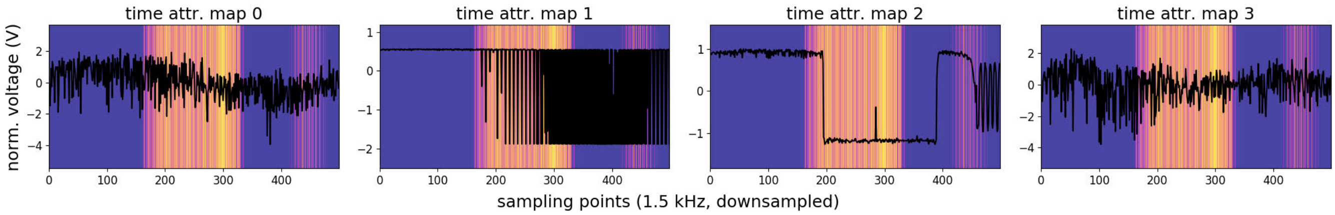


FIGURE 6. Time attribution maps of four synchronously recorded signals using real-world automotive sensors.

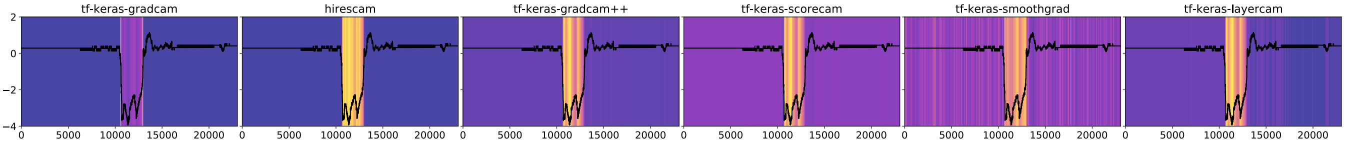


FIGURE 7. Saliency map generation methods side-by-side [20].

various signals, it is of interest to consider the *time attribution maps* in Fig. 6, i.e., points in time at which interactions of interest took place. Both types of attribution map are obtained from the XCM [59] architecture.

VI. NEURO-SYMBOLIC ANOMALY DETECTION AND FAULT DIAGNOSIS

As the previous sections have shown, knowledge- and machine-learning-based diagnosis requires the integration of various components. To define the prototypical overall process (*diag. circuit* in Fig. 1) and to integrate all developed modules, a state machine was defined.[4] It is a domain-agnostic generalization of the *vehicle diagnosis state machine* from [20]. Fig. 8 shows its architecture, which is implemented using the *smach*[5] library. Initially, there is meta and context data processing. Based on the read information, the entity-specific instance data is entered into the KG (cf. Sec. IV-D). If error code data is available, the KG is extended with the processed data, i. e., the information that the fault conditions represented by the individual error codes occurred in the respective entity, etc. If the entity instance already exists in the KG, it is extended, otherwise it is newly created. Based on the acquired fault context, the actual diagnostic process is initiated.

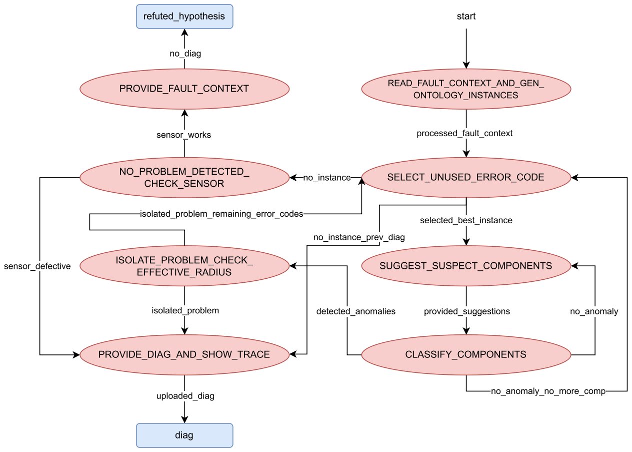


FIGURE 8. Domain-agnostic neuro-symbolic diagnosis state machine (generalizing [20]).

The actual diagnosis starts with a state in which a best-suited error code instance is selected for further processing. There are two possible transitions. If an instance is selected, the process continues with suggesting suspected components. Otherwise, no error was detected and the indirect conclusion of a potential sensor malfunction is provided. This conclusion should be verified or refuted by the human. Then there is one of two outcomes: Either the sensor works, which means that the diagnosis was unsuccessful

[4] https://github.com/tbohne/nesy_diag_smach/releases/tag/v0.1.6

[5] Python library to build hierarchical state machines [67].

(*refuted_hypothesis* in Fig. 8, only the disproved initial hypothesis and the context are provided due to unmanageable uncertainty), or there is a diagnosis of a defective sensor (cf. Fig. 8). The case of an unsuccessful diagnosis is not a weakness of the system, but rather the handling of the edge case in which a potentially existing previous error or anomaly is no longer present in the diagnosed entity, i.e., there simply is nothing to diagnose. The more interesting case: Certain components in the entity of diagnosis are recommended to be investigated in light of the available information (fault context). Based on the selected error code instance, the *KG query tool* is used to query the corresponding suspect components, and for each, whether it can be reasonably diagnosed with the considered sensor. Afterwards, we first distinguish between the subset of suspect components for which sensor diagnosis is appropriate and those that must be verified manually. Then, synchronized *sensor recordings* are performed at the proposed components of the respective subset, and the resulting *time series are classified* using trained ANN models (cf. Sec. V). The prediction can be interpreted by *overlaid heatmaps* (cf. Sec. V-A). Subsequently, the subset of recordings to be inspected manually is handed over to the human. In the end, there is a set of anomalous components identified by the trained models and the human. If this set is empty, i. e., no anomaly has been detected, the next iteration of suggestions follows. However, if no anomaly is detected and there are no remaining components to suggest, the next error code instance is selected. If anomalies are found, though, the *root cause analysis* follows, which is explained in Section VI-A. All the specifics and subtleties of the diagnostic procedure can be read in [20].

A. ROOT CAUSE ANALYSIS TO DETERMINE THE SOURCE OF THE DEFECT

Once an anomaly is identified in the described manner, the fault is isolated by recursively inspecting the cause-effect relationships in the entity of diagnosis, which are part of the KG (cf. Sec. IV-A), i. e., graph traversal coupled with anomaly detection. This basically creates a causal sub-graph for each anomalous component from which the root cause can be derived. After all, in technological systems errors are rarely encountered that are entirely independent of other components in the system. Typically, there are cascading paths where a problem starts at one component and then spreads to others. The actual interest is directed at the root cause instead of mere side conditions. The termination criterion is that there are no further known components that have not yet been examined and that could directly affect an anomalous component. During the recursive procedure, the fault path, explicitly considered links, etc. are tracked. The result is visualized dynamically for each diagnosis, e. g., Fig. 9.

The recursive and dynamic sub-graph construction based on KG queries is defined by the pseudocode shown in Algorithm 1 [20], initialized with an empty graph and the initial anomalous component C_D. After isolating the problem, the diagnosis is entered into the KG, along with a detailed record of all relevant information that led to it (cf. Sec. IV-B), to learn from it and facilitate future diagnoses. Ultimately, the diagnosis is presented in the form of the inverted fault path in Fig. 9 as this was the *affected-by*-direction, not the direction starting from the probable root cause of the fault. For the example in Fig. 9, this would be $\{C_B \rightarrow C_A \rightarrow C_D\}$. So the problem probably started at C_B, cascaded through C_A, and finally to C_D. Although it is not guaranteed to be the actual root cause, it should provide a domain expert with a fairly strong understanding of the problem prevailing in the entity in question.

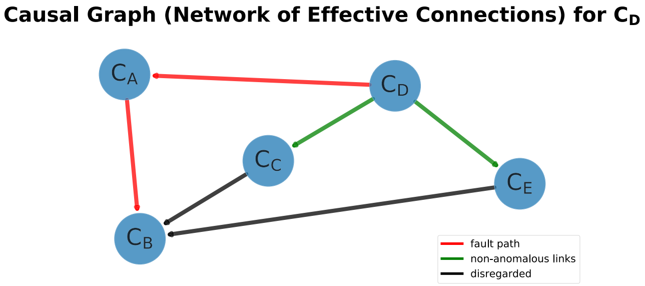


FIGURE 9. Fault isolation result example [20].

Algorithm 1 Recursive Function That Constructs the Complete Causal Graph for the Specified Components

```
Input: graph: dictionary, components: list
Output: constructed causal graph
if len(components) == 0 then
    return graph
end if
comp ← components.pop(0)
if comp not in graph.keys() then
    affecting_comp ← query_affected_by(comp)
    components ← components ∪ affecting_comp
    graph[comp] ← affecting_comp
end if
return construct_causal_graph(graph, components)
```

B. NEURO-SYMBOLIC CIRCUIT

The suggested components to measure during diagnosis stem from the KG. After the signals are recorded, they are interpreted and a heatmap is overlaid (cf. Fig. 7). A crucial idea of the overall approach is to close the circuit and feed this information back into the KG. In case of an anomaly, it is the information where the error is located in the signal, i. e., generally where the system tells us to look to identify the problem under consideration, the region of interest (ROI). This is a very useful debugging resource, highlighting issues such as overfitting and deviation from expert judgements. If the classification of a whole range of signals reveals that, for instance, certain highlighted segments correlate with certain error code ranges, this is very valuable knowledge that is unavailable a priori. This requires reliable and accurate heatmaps, which is why we put so much emphasis on comparing different well-established techniques for generating them in [20]. Since plausibility may also depend on the context of the classification, the interpretable symbolic state transitions etc. are crucial information for the assessment. For experts, it is often particularly difficult to precisely specify the properties of such a ROI. It is often a very intuitive, experience-based and hard to grasp process, so it could be very valuable to simply learn the ROIs in this way and then compare them to the intuitive notions of human experts. There may also be patterns that are very subtle and difficult for humans to recognize. Thus, the classification, i. e., the neural part of the neuro-symbolic system, benefits from the KG, which essentially narrows down the search space, and the KG in turn benefits from the neural part, namely from the results and explanation of the ANN-based classification. In the end, if a threshold is exceeded for an error code, i. e., a certain number of roughly agreeing heatmaps has been gathered for it, one could crop this sub-ROI and train a classification model for it. Once this error occurs, the sub-ROI is cropped and the more specialized model is applied, resulting in a sub-ROI patch classification. In conclusion, the system theoretically gets better at diagnosing errors that it has seen frequently in the past. A further option is to cluster all recorded heatmaps, irrespective of the particular fault, in order to find patterns. This is only one illustration of the opportunities for *knowledge discovery*. As introduced in Section IV-B, all kinds of relevant diagnostic information are gathered and linked so that previously unknown correlations can be discovered by deploying the system in practice. A thorough demonstration and evaluation of the capability of knowledge discovery is beyond the scope of this work. However, in a separate work, empirical results on well-established time series classification datasets demonstrate the effectiveness of our saliency map-driven method for knowledge discovery [68].

VII. SYSTEMATIC EVALUATION

This section focuses on a systematic evaluation of the architecture using randomized, parametrized synthetic problem instances and corresponding ground truth solutions generated based on the formalism established in the following evaluation framework.[6] It enables a quantitative approach even in cases where there is a lack of sensor data and expert knowledge. Hence, it also extends previous work [20], in which only a qualitative discussion took place.

In general, even under perfect conditions, it is not possible to collect an unlimited amount of data and knowledge that covers all possible scenarios and enables a truly comprehensive evaluation of the diagnostic system. As [69] correctly states, hybridization can take many shapes, which poses a particular challenge for common benchmarks. In the following, we set up a benchmark that goes beyond the specific use case in [20] by formalizing and generalizing the core properties that it shares with many other diagnostic domains. In a sense, we explore the performance of the system across a broad spectrum of configurations and thus theoretical domains.

While the architecture is presented as a human-in-the-loop system, the systematic evaluation is fully automated, since the role of the human is not crucial for assessing the performance and limitations of the system itself. We merely skip the waiting times for human interactions when providing input, etc. From the perspective of the system, it makes no difference. The only substantial difference lies in human interventions, such as doubting classifications or entire diagnoses based on the explanatory report. While evaluating this would constitute a substantial contribution, it is clearly beyond the scope of this work, which is focused on evaluating the functioning of the system, not the subtleties of human-machine interactions and interpretations. In the context of the present evaluation, it is assumed that the human provides input instantaneously and that all suggestions are accepted. In practice, as previously discussed, it can be beneficial to repeat classifications, but there is no mechanism yet to automatically improve false recommendations (beyond logging). This is future work. The human role in terms of efficiency or runtime is reflected in the following sections.

Table 1 summarizes the notations of the parameters and metrics that are introduced in the following sections. In order to facilitate understanding of how the abstract parameters used in the synthetic instances map to the characteristics of real-world diagnostic domains, and to assess the practical relevance of the evaluation's conclusions, the corresponding concept of the general ontology in Fig. 2 is mentioned for each. All of these are substantiated by corresponding real-world equivalents in the automotive domain demonstrated in [20]. Let C be the set of components (`SuspectComponent` in Fig. 2) of constant size that are part of a diagnostic domain. Each diagnostic problem instance is composed of $|C| := 129$ components. This is motivated by two aspects: First, we plan to experiment with the 129 UCR datasets[7] in the future. Each dataset represents one component, this is already anticipated here. Additionally, based on the concrete instantiation of our system in [20], this seems to be a reasonable order of magnitude for many practical problems. Moreover, each instance specifies which components have anomalies (underlying ground truth solution for `prediction` of `Classification` in Fig. 2), with which other components they are causally related (`affected_by` in Fig. 2), the ground truth fault paths (expected diagnosis, underlying ground truth solution for the predicted `FaultPath` in Fig. 2), the input error codes (`ErrorCode` in Fig. 2) and the corresponding suspect components (`DiagnosticAssociation` in Fig. 2), and finally the simulated accuracies for the classification

[6] https://github.com/tbohne/nesy_diag_bench/releases/tag/v0.0.2

[7] https://www.cs.ucr.edu/ eamonn/time_series_data_2018/

TABLE 1. Notations of parameters and metrics.

Variable	Description
C	set of components of constant size
I	set of instance sets
$i \in I$	instance set
$i_\eta \in i$	specific instance from instance set
α	anomaly percentage
β	*affected-by* percentage UB per component
γ^{LB}, γ^{UB}	LB, UB for ANN model accuracy
δ	UB percentage for *distractors*
ϵ	UB for fault path component percentage
ζ	seed for random generation processes
η	instance index
$p_0^{i_\eta}$	anomaly link score
$p_1^{i_\eta}$	ground truth fault path score
p_2^{i}	ground truth match percentage
$p_3^{i_\eta}$	anomaly score
f_i^a, f_i^{max}	average, maximum number of fault paths
l_i^a, l_i^{max}	average, maximum fault path length
$\tilde{f}_i^a, \tilde{f}_i^{max}$	average, maximum fault path deviation
$c_1^{i_\eta}$	compensation by *affected-by savior*
$c_2^{i_\eta}$	missed compensation chances
$c_3^{i_\eta}$	*no second chance* cases
d_s^i	diagnosis success percentage
$c_r^{i_\eta}$	classification ratio
$n_c^{i_\eta}$	number of classifications
r_i^a, r_i^m, r_i^{max}	average, median, maximum runtime
TP, TN, FP, FN	actual classification results
t_p, t_n, f_p, f_n	expected classification results
n_p, n_n	ground truth number of positives and negatives
$m_1^{i_\eta}$	missed anomalies (misclassifications)
$m_2^{i_\eta}$	missed anomalies (unclassified)
$m_3^{i_\eta}$	all missed anomalies

models (`Model` in Fig. 2) of each component. Overall, the following results are based on the systematic solving of 4000 random-generated, parametrized diagnostic problem instances, as described below. The number of instances arises from the possible combinations of the defined parameter intervals subtracted by some parameter combinations that do not yield any further insights. The aim of this evaluation is twofold: We show that the proposed system functions as expected, but we also provide insights into the structure and properties of the considered general diagnosis problem.

First, we introduce some variables: α: *anomaly percentage*, β: *affected-by percentage* (upper bound), and γ: *model accuracy*. Depending on the context, the variables are either denoted in fractional representation ([0, 1]) or as explicit percentages. α represents the fraction of C that comprises anomalous components (randomly generated and uniformly distributed), β specifies an upper bound (UB) for the random *affected-by* links of each component, and γ is self-explaining. The ground truth fault paths are generated based on α and β. We consider the component network and recursively follow the *affected-by* links of all anomalous components. These are the edges of the final fault paths. Only unique, longest fault paths are taken into account, i. e., sub-paths are ignored. Furthermore, let δ be the UB percentage for *distractors*. Distractors are suspect components that are not part of the ground truth fault path, but nevertheless associated with some input error code. Moreover, let ϵ be the UB for the *fault path component percentage*, based on which diagnostic associations are generated. An input error code is generated for each ground truth fault path, and each error code should have a number $[1, \lfloor \frac{\epsilon}{100} n \rfloor]$ of randomly associated components from the corresponding ground truth fault path of length n. There have to be as many random error codes as there are ground truth fault paths (assuming no duplicates). The first must always be the "anti root cause", i. e., the start of the *affected-by* chain, so that all components are reachable and diagnosis is feasible. Both δ and ϵ make the problem a little more challenging and realistic. Not all ground truth anomalies are already indicated by the error code, and there are also some indicated components that turn out to be irrelevant. With distractors, we are essentially assuming that the error code association to components is not perfectly correct and relevant in each case. Generally, it would be possible to count how often a distractor leads to a false positive (incorrectly identified anomaly). However, this is not of interest because there is no fundamental difference between a distractor or any other component during the diagnosis leading to a false positive. The distractors are only concerned with increasing the likelihood of false positives. The specific components that function as distractors in a particular case are not significant for the evaluation. Finally, let ζ and η be the seed for the random generation processes and the index of each instance in the set, respectively. Each instance set $i \in I$ is composed of 100 instances $i_\eta \in i$, randomly generated based on the same configuration that is coded in the instance name, i. e., $< |C|\_\alpha\_\beta\_\epsilon\_\delta\_\gamma^{LB}\_\gamma^{UB}\_\zeta\_\eta >$.[8]

Fig. 10 shows an abstract overview of the evaluation process. It starts in the top center cell by specifying the parameters of the instance set to be evaluated. Based on this, the randomly generated instance (JSON) and the corresponding knowledge graph (*n*-triples) are generated and serve as input, background knowledge and ground truth solution for each generated problem instance. Cell three sketches one of the main aspects of the background knowledge, i.e., the components of the diagnosis entity together with their causal connections. Subsequently, the state machine solves the provided instance and presents the

[8] https://github.com/tbohne/nesy_diag_bench/blob/main/res/exp_instance_sets.zip

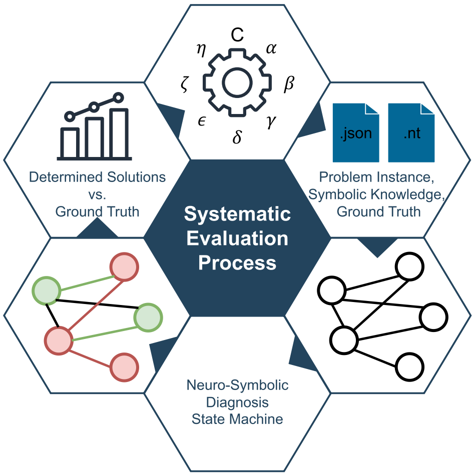
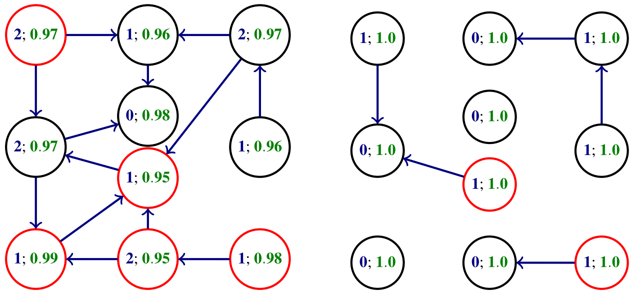


FIGURE 10. Abstract vis. of the evaluation process.



(a) $\alpha = 0.5$, $\beta = 0.2$, $\gamma \in [0.95, 0.99]$

(b) $\alpha = 0.2$, $\beta = 0.1$, $\gamma^{LB} = \gamma^{UB} = 1$

FIGURE 11. Effects of α, β, γ^{LB} and γ^{UB} for $|C| = 10$.

diagnosis in cell five. Finally, in the last cell, the determined solutions are compared to the ground truth solutions of the instances.

The underlying KG is automatically replaced per instance during evaluation. The most important aspects that are part of the KG files generated for each random problem instance, i. e., for each n-triples file, are: suspect components, *affected-by* relations, input error codes, diagnostic associations between error codes and components, priority IDs of diagnostic associations, and finally fault conditions. For each instance, it is ensured during instance generation that it is actually solvable. Some of the configurable parameters are not changed throughout the experiments, because they are not as important for the analyses and fixated to practically plausible default values: $|C| = 129$ (cf. above), $\epsilon = 0.5$, $\delta = 0.1$. Thus, we will use the following notation to precisely refer to the most important (varying) aspects of the instance set configuration: $< \alpha\_\beta\_\gamma^{LB}\_\gamma^{UB} >$. The graphs in Fig. 11 illustrate the effects of the parameters α, β and γ on the problem structure. The nodes represent components of the diagnosis entity and the edges represent the *affected-by* relations between components. Red nodes represent anomalies, and the out-degree is annotated at each node along with the accuracy of the corresponding trained neural network model (*out-degree; model accuracy*). β sets an UB for the out-degree of each node and $[\gamma^{LB}, \gamma^{UB}]$ defines the interval for the simulated uniformly distributed random model accuracies. Thus, we see 50% anomalies, model accuracies in $[0.95, 0.99]$, and an out-degree of at most 2 in Fig. 11a. Reducing α and β (cf. Fig. 11b) leads to 20% anomalies and an out-degree of less than or equal to 1. In this case, the model accuracy is always 1. The results of the following sections are based on systematically generated instance sets based on the following intervals for the varying parameters: $\alpha \in \{0.01, 0.05, 0.1, 0.2\}$, $\beta \in \{0.01, 0.02, 0.03, 0.05, 0.07, 0.1, 0.2\}$, $\gamma^{LB} \in \{0.9, 0.95, 1.0\}$, and $\gamma^{UB} \in \{0.95, 0.99, 1.0\}$. All parameters considered are either set to practically plausible, domain-agnostic values or intervals were specified, the limits of which are empirically justified below. The values for γ^{LB} and γ^{UB} are always set in pairs, i. e., they always share the same index in the set of configurations. These synthetic instances are abstract representations of the specific problem formulation in [20] as well as many other conceivable practical domains.

The following section will analyze the impact of certain configuration aspects on the solving procedure and results. Instead of only visually estimating the correlation, we compute the *Pearson correlation coefficient* ρ for many plots in the following. As always, this is not causation. However, it is an approximation that answers the question of whether causation is possible, i. e., an indicator used to confirm or refute certain hypotheses. We expect a classifier, e.g., a trained neural network model, for each component $c \in C$ and we know the accuracy γ of each (cf. node annotations in Fig. 11). The accuracy on new, unseen data is not necessarily converging to the accuracy of the models. This depends on the dataset distribution, i.e., data drift, model generalization capabilities, etc. It is not guaranteed that a model performs similarly to the established accuracy on unseen data. However, as our objective is not to evaluate the trained classifiers but the diagnosis system as a whole, in the evaluation, we do not train neural networks; rather, we assume they have a certain known accuracy. Then, we simulate the application of neural networks by generating random outcomes with a probability of being correct based on a model's expected accuracy. Consequently, over a large number of trials, the simulation's accuracy should converge to the model's original accuracy due to the *law of large numbers*. Accordingly, over a sufficiently large number of simulated runs, the resulting accuracy should be approximately equal to the model's accuracy, assuming the outcomes are independent.

The evaluation is contingent on the assumption that the dataset the models are applied to exhibits similar characteristics to the original test data from which the accuracy value was derived and sort of ignores the inherent variability in real-world data. Nevertheless, the simulation is valuable for exploring theoretical outcomes over a wide range of problem configurations, i. e., domains. It is permissible to disregard this kind of variability here because it is already factored into the accuracy spectrum and the evaluation takes place at a higher level. The aim here is not the typical performance evaluation of neural networks, but rather the evaluation of the overall system, given a spectrum of trained neural networks with certain known accuracies, in order to assess the extent to which the system can be effectively applied to a problem scenario when a trained model of a certain quality is available. Actually training and applying neural networks would not contribute anything to the present evaluation beyond proving the existence of such models, which is unquestioned. In general, we consider a binary classification scenario in which the probability for the positive class is lower than that for the negative class, i.e., an imbalanced class distribution. However, it is only imbalanced in terms of the expected occurrences in practice, not in terms of training data.

A. INSTANCE LEVEL RESULTS

In order to discuss results on the instance level, we show the solutions of $i =< 129\_10\_20\_50\_10\_95\_99\_42 >, i \in I$ as one representative, instructive instance set. Fig. 12 shows the fault path distribution across the 100 instances $i_\eta \in i$. There is a positive correlation of $\rho = 0.84$ ($p \ll 0.001$) between the number of fault paths and the average fault path length (cf. third plot in Fig. 12), i. e., with increasing fault path length, the number of fault paths also increases. This makes sense, as there are more permutations with more components per fault path. Of course, all permutations could only be possible for $\beta = 1.0$, which would imply a fully connected anomaly graph. Usually, due to β, not all permutations are feasible. Nevertheless, there are more feasible permutations with larger α and β values. For a fully connected graph, there are $(\alpha|C|)!$ permutations. The question is how many permutations are to be expected based on values $\beta \neq 1.0$. For this particular instance set, there are $(0.1 \cdot 129)! = 6.23e^9$ total permutations, assuming a fully connected anomaly graph. However, the median number of ground truth fault paths is 13.5, which represents $2.17e^{-9} \approx 0\%$ of the total. It is possible to approximate the number of permutations that are actually feasible based on the input parameters of the problem domain (α, β, C), though:

$$l_i^a \approx \lfloor \frac{\log\left((\lfloor \alpha|C| \rfloor - 1)\frac{\beta}{2}\right)}{\log\left(\frac{1}{1-\frac{\beta}{2}}\right)} \rfloor \tag{1}$$

$$f_i^a \approx \left(\sum_{j=1}^{l_i^a} \frac{\beta}{2}(\lfloor \alpha|C| \rfloor - j)^2\right) + \lfloor \alpha|C| \rfloor \tag{2}$$

First, in (1), i. e., in the estimated fault path length, $(\lfloor \alpha|C| \rfloor - 1)\frac{\beta}{2}$ represents the expected number of anomalous connections, i. e., the number of anomalous links that each anomaly is expected to have. Consequently, the remaining anomalies are multiplied by the branching factor. The logarithm is introduced to account for diminishing or non-linear growth. For larger networks (α) or higher connectivity (β), fault paths tend to be longer, but the growth rate is non-linear. The denominator $\frac{1}{1-\frac{\beta}{2}}$ reflects how the connectivity of the network determines the expected path length. A higher degree of connectivity results in longer potential fault paths. When β is small, i. e., when considering a sparse network, $1 - \frac{\beta}{2}$ is slightly smaller than 1, and $\frac{1}{1-\frac{\beta}{2}}$ is only slightly larger than 1, indicating limited growth. Whereas for a highly connected network, i. e., β close to 1, $1 - \beta$ approaches 0 and the total fraction becomes large, reflecting the higher number of connections and thus longer expected fault paths. Therefore, the denominator essentially scales the growth rate based on connectivity.

In (2), the sum represents the number of levels in the recursive fault path generation process. In each iteration, it adds an estimate of the number of new fault paths generated at that level. Thus, at each level j, every anomaly is multiplied by the branching factor and its remaining potential, i. e., the number of anomalies subtracted by j. The currently considered anomalies are multiplied by the remaining anomalies, where current and remaining are approximately equal for one iteration. Thus the square. At each later level of the procedure, there should be fewer anomalies available to connect to. The first layer represents the anomalies $\lfloor \alpha|C| \rfloor$ in the first iteration of the sum. This is multiplied by the branching factor $\frac{\beta}{2}$, which represents how many anomalies each anomaly is connected to. However, these anomalies are again connected to the remaining number of anomalies, i. e., $\lfloor \alpha|C| \rfloor - 1$ for each branch. The -1 is dominated by the $-j$, which is why it is sufficient to square the number of anomalies subtracted by j and multiply it by the branching factor in each iteration. Therefore, the number of expected paths per level considers pairwise combinations of remaining anomalies. It is an estimate of the feasible permutations of each anomaly, along with all of its expected associated anomalies, added to the initial number of anomalous components, e. g., to reflect entirely isolated components.

The diagnosis system filters out redundant fault paths, e. g., fault paths that are a subset of others. Also, adding $\lfloor \alpha|C| \rfloor$ in (2) can be too high, in the most extreme counterexample all anomalies could be part of a single fault path, no isolated ones. Nevertheless, the other extreme case is just as possible, whereby all anomalies are isolated. Therefore, the approximation could generally slightly overestimate the number of fault paths, yet it should still be a reasonable approximator for the expected order of magnitude. Each anomalous component is at least part of one fault path, possibly more based on β. β has to be divided by 2 because it is an UB. The actual values will be equally distributed in $[0, \beta]$. In this case, the approximation yields:

$$l_i^a \approx \lfloor \frac{\log\left((\lfloor 0.1 \cdot 129 \rfloor - 1)\frac{0.2}{2}\right)}{\log\left(\frac{1}{1-\frac{0.2}{2}}\right)} \rfloor = \lfloor 0.90 \rfloor = 0 \tag{3}$$

$$f_i^a \approx \left(\sum_{j=1}^{0} \frac{0.2}{2}(\lfloor 0.1 \cdot 129 \rfloor - j)^2\right) + \lfloor 0.1 \cdot 129 \rfloor = 12 \tag{4}$$

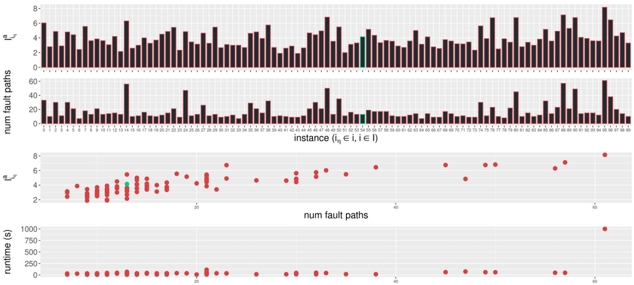


FIGURE 12. Fault path distribution for $i \in I$.

Accordingly, it estimates the order of magnitude of the expected number of fault paths rather well, in this case the true value is ≈ 112.5% of the estimated value. However, this is a somewhat uninteresting case since the sum in (4) has no influence due to the very small estimated fault path length of 0.9. To provide a more meaningful example, we consider a configuration with $\alpha = 0.3$ and $\beta = 0.1$:

$$l_i^a \approx \lfloor \frac{\log\left((\lfloor 0.3 \cdot 129 \rfloor - 1)\frac{0.1}{2}\right)}{\log\left(\frac{1}{1-\frac{0.1}{2}}\right)} \rfloor = 11 \tag{5}$$

$$f_i^a \approx \left(\sum_{j=1}^{11} \frac{0.1}{2}(\lfloor 0.3 \cdot 129 \rfloor - j)^2\right) + \lfloor 0.3 \cdot 129 \rfloor \approx 643 \tag{6}$$

The median ground truth number of fault paths in this case is 804.5, demonstrating the appropriateness of the approximation (≈ 125% of the estimated value), even with larger estimated fault path lengths of $l_i^a = 11$ in this case. The corresponding ground truth average fault path length is 10.9, i. e., the approximation of l_i^a also works reasonably well, in this case almost perfectly. Additionally, Fig. 12 shows that most instances lead to relatively low solving runtimes, regardless of the number of ground truth fault paths, with one exception: Instance $\eta = 95$ leads to a runtime of 1000.9 s. As can be seen in the first two plots, this is also the one with the most and longest fault paths, but the difference in terms of runtime is far greater compared to the other two attributes. For this instance, it is of interest to consider another metric, the *fault path deviations* $\tilde{f}_i$, which is defined as the absolute difference between the number of determined fault paths and the number of ground truth fault paths (f_i). This instance has a huge deviation of $f_{i_\eta} = 3991$.

B. CUMULATIVE RESULTS

The following section discusses cumulative results,[9] i. e., aggregated results across the 100 randomized instances generated per instance set $i \in I$. Let $G_{i_\eta} := (C, E_{i_\eta})$ be the causal graph for some instance $i_\eta \in i$ of some instance set $i \in I$, i. e., the network of effective connections, and $A_{i_\eta} \subseteq C$ the set of anomalous components. Furthermore, $\hat{A}_{i_\eta} \subseteq E_{i_\eta}$ represents the ground truth anomaly links with $\hat{A}_{i_\eta} := \{(v_1, v_2) \in E_{i_\eta} \mid (v_1, v_2) \in A^2_{i_\eta}, v_1 \neq v_2\}$ and F_{i_η} the result of a diagnosis run for instance i_η, i. e., a determined set of fault paths. Finally, let $\tilde{F}_{i_\eta}$ be the set of ground truth fault paths. We then define the set of determined anomaly links as $\hat{F}_{i_\eta} := \{(c_n, c_{n-1}), (c_{n-1}, c_{n-2}), \ldots, (c_{n-(n-1)}, c_0) \mid c_i \in f \,\forall f \in F_{i_\eta}\}$. Based on this, we define $p_0^{i_\eta}$ as the *anomaly link score*, $p_1^{i_\eta}$ as the *ground truth fault path score*, p_2^i as the *ground truth match percentage*, and $p_3^{i_\eta}$ as the *anomaly score* as key performance metrics:

$$p_0^{i_\eta} := \begin{cases} \frac{|\hat{F}_{i_\eta} \cap \hat{A}_{i_\eta}|}{|\hat{A}_{i_\eta}|} & \text{if } |\hat{A}_{i_\eta}| > 0 \\ 1.0 & \text{else} \end{cases} \tag{7}$$

$$p_1^{i_\eta} := \frac{|F_{i_\eta} \cap \tilde{F}_{i_\eta}|}{|\tilde{F}_{i_\eta}|} \tag{8}$$

$$p_2^i := \sum_{i_\eta \in i} 1_{\{\tilde{F}_{i_\eta} = F_{i_\eta}\}} \tag{9}$$

$$p_3^{i_\eta} := \frac{\alpha|C| - (\alpha|C| - TP)}{\alpha|C|} = \frac{TP}{\alpha|C|} \tag{10}$$

Fig. 13 primarily serves as a motivation to not exceed α and β values of 0.2, i. e., to only consider $\alpha, \beta \in [0.0, 0.2]$. The *classification ratio* $c_r^{i_\eta}$, i. e., the fraction of components $c \in C$ of the diagnosed system that are classified, is already approaching 1 for some $i \in I$. In such cases, it basically performs an exhaustive search, i. e., it simply checks all components, which is not a sensible scenario for the system. $c_r^{i_\eta}$ obviously depends on the combination of α and the connectivity β. As can be seen in Fig. 13, there is a general tendency for increasing $\bar{c_r}^i$ to result in worse ground truth match percentages p_2^i, with the exception of cases where the model accuracy is extremely high or even perfect. The conclusion of this plot is that it is advantageous to minimize $c_r^{i_\eta}$ or to pay close attention to obtain very accurate models, i. e., γ values close to 1. Generally, FPs (regular components treated as anomalies) lead to unnecessary additional classifications based on their *affected-by* relations. This can result in an overall increased $c_r^{i_\eta}$. Conversely, FNs (missed anomalies) can reduce the $c_r^{i_\eta}$.

The number of FPs naturally depends on three variables: α, β, and γ (cf. Fig. 14). α in isolation is not able to increase the number of FPs. In fact, it can even reduce the potential for them because there are more TPs. However, in combination with β it leads to larger $c_r^{i_\eta}$, which also only lead to FPs if γ allows for misclassifications. The number depends on how far γ is away from 1.0 and how many classifications are performed based on α and β. Even when the value of γ is relatively low, the number of FPs remains relatively low when there are few classifications based on α and β. On the other hand, if α and β are both exceedingly high, and γ is perfect, then α and β have no influence whatsoever, the number of FPs will always be 0. Therefore, the outcome depends on the interplay of the three. As anticipated and also visible in Fig. 14, α in isolation is not able to fully

[9] https://github.com/tbohne/nesy_diag_bench/blob/main/res/exp_solutions.zip

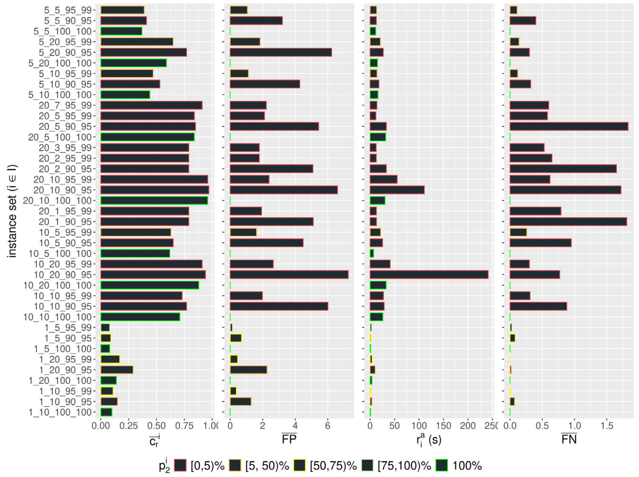


FIGURE 13. Classification ratio.

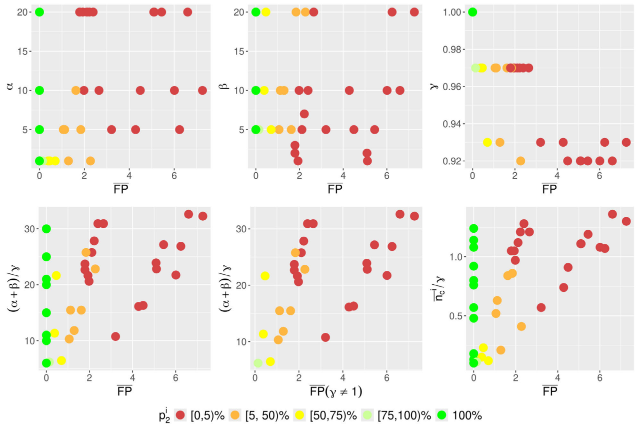


FIGURE 14. FP analysis.

explain the number of FPs, e. g., $\alpha \in [5, 20]$ covers the entire FP range on the x-axis, although a significant positive correlation is visible ($\rho = 0.35, p \ll 0.05$) due to increased $c_r^{i_\eta}$. β in the second plot is a way worse predictor with no correlation at all. The third plot shows an expected strong negative correlation of $\rho = -0.84$ ($p \ll 0.001$) between γ and the number of FPs. This is because γ is causal in each case, for all values of α and β. The parameters α and β solely determine the number of applications of γ. It may also be worthwhile to aggregate the variables and analyze the combined correlation with the number of FPs. The weighted average $\frac{w_1\alpha+w_2\beta+w_3\gamma}{\sum_i w_i}$ yields $\rho = 0.39$ ($p \ll 0.05$), with weights conforming the aforementioned intuitive explanation of setting a focus on γ and rating α as slightly more important than β, i. e., $w_1 = 0.5, w_2 = 0.3, w_3 = 1.0$. The fourth plot aggregates the three variables as $\frac{\alpha+\beta}{\gamma}$, resulting in $\rho = 0.44$ ($p \ll 0.05$). Except for β in isolation, there is a quite significant correlation in all cases, indicating that the number of FPs indeed correlates with these three variables. It is also reasonable to exclude cases of 100% accurate models, as this eliminates the influence of the other two variables (α, β). The fifth plot shows the filtered results with $\rho = 0.56$ ($p \ll 0.05$). Finally, the sixth plot correlates the number of FPs with the ratio $\frac{\bar{c_r}^i|C|}{\gamma}$, which leads to $\rho = 0.55$ ($p \ll 0.001$). This provides insight into how the number of classifications affects the FPs in relation to γ. Obviously, this would also be higher when excluding the 100% cases.

Thus, γ in isolation is the variable that exhibits the strongest (negative) correlation with the number of FPs. This effect would be even stronger if the range $[\gamma^{LB}, \gamma^{UB}]$ were to be increased. Obviously, even when the number of FPs fully depends on the values of α, β, and γ, it is not necessarily possible to find a simple aggregation function for them that results in a perfect correlation, given that their relationship may not be strictly linear. Each entry, i. e., each instance set $i \in I$, in the plots is color-coded with its p_2^i performance. Essentially, how many instances were solved entirely correctly, without a single misclassification.

The number of FNs intuitively depends on the anomaly percentage and the model accuracy, i. e., α and γ, as is confirmed in Fig. 15.

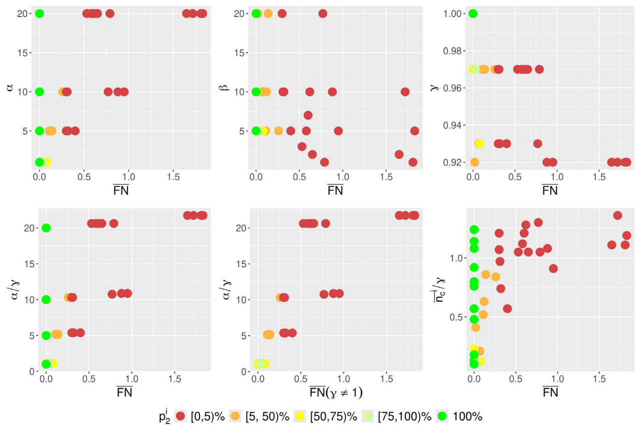


FIGURE 15. FN analysis.

The first plot shows a clear correlation between α and the number of FNs ($\rho = 0.65, p \ll 0.001$). It is a lot stronger than the correlation between α and the number of FPs, which was $\rho = 0.35$. This is intuitively reasonable, since anomalies, which are determined by α, are a precondition for FNs, but not for FPs. As anticipated above, β should be less relevant for FNs, which is also confirmed in the second plot. Nevertheless, there is a significant negative correlation of $\rho = -0.36$ ($p \ll 0.05$) with β as opposed to the FP case, which may appear surprising. This can be attributed to the fact that FNs circumvent the usage of *affected-by* links by falsely triggering the termination criterion. One might expect that the amount of β should have no influence on the number of FNs. As mentioned above, FNs intuitively depend on the number of anomalies (precondition) and the model accuracy. Higher β can generally support an increased $c_r^{i_\eta}$, and more classifications can also lead to more misclassifications of anomalies and thus more FNs. However, the exact opposite is the case, namely a negative correlation. The FNs increase with decreasing β. The effect can be explained as follows: At lower β, the FNs deactivate fewer links. Deactivating links would reduce the $c_r^{i_\eta}$, and thus the $c_r^{i_\eta}$ is not reduced at lower β, which also preserves the chance for FNs. However, a lower β also reduces $c_r^{i_\eta}$ by itself. This would therefore mean that the deactivation effect of FNs is stronger than the increased $c_r^{i_\eta}$ due to higher β. A lower β particularly leads to situations in which many anomalies are separated from one another and thus only reachable via input error code and not via *affected-by*. Then an FN does not prevent other FNs from occurring due to the termination criterion. The third plot again illustrates a crucial relation. There is a strong negative correlation of $\rho = -0.66$ ($p \ll 0.001$) between γ and the number of FNs, which is comprehensible given that higher model accuracies prevent FNs. Due to the fact that β seems less relevant for the number of FNs (more than for the number of FPs, but still less than the other two variables), we only aggregate α and γ, i. e., $\frac{\alpha}{\gamma}$, in the fourth plot, which shows a very strong correlation of $\rho = 0.69$ ($p \ll 0.001$). Once again, the filtered version in the fifth plot shows an even higher correlation of $\rho = 0.78$ ($p \ll 0.001$). Finally, for previously discussed reasons, $\frac{\bar{c_r}^i|C|}{\gamma}$ shows about as much ($\rho = 0.58, p \ll 0.001$) positive correlation as in the FP case.

1) RUNTIME ANALYSIS

Let r_i^a be the average runtime, r_i^m the median runtime, and r_i^{max} the maximum runtime of some instance set $i \in I$. Furthermore, let f_i^a represent the average number of fault paths, f_i^{max} the maximum number of fault paths, and $\tilde{f}_i^a$, $\tilde{f}_i^{max}$ the analogue for the fault path deviations of some instance set $i \in I$. Moreover, let l_i^a be the average and l_i^{max} the maximum fault path length. r_i^a is not significantly correlating with $\bar{c_r}^i$ (cf. Fig. 13). At first glance, it may appear to be due to the outliers in e. g., $i =< 10\_20\_90\_95 >$. However, an examination of r_i^m reveals that there are slight changes, but the overall structure remains consistent. There is no clear correlation between $\bar{c_r}^i$ and the runtime. Therefore, the number of classifications is not the dominant factor in determining longer runtimes. The runtime correlates quite well with the number of fault paths and their length (cf. Fig. 16). Interestingly, $r^a_{<10\_20\_90\_95>}$ and $r^m_{<10\_20\_90\_95>}$ are by far the longest (cf. Fig. 13). As shown in Fig. 17, there are some extreme outliers in this instance set in terms of runtime. Fig. 17 also illustrates that those perfectly coincide with the outliers in terms of fault path deviations.

Fig. 18 is a highly insightful plot for $i =< 20\_10\_90\_95 >$. First, in this plot, it is evident that with $f_i^{max} = 2146$ there is one extreme outlier in terms of the number of fault paths (instance $\eta = 74$). However, crucially, this fault path outlier does not correspond to the runtime outliers, as can be seen in the fourth plot. It only has a runtime of 223.3 s. In contrast, the runtime outliers ($\eta = 49 : 2806.1\ s$, $\eta = 81 : 1609.6\ s$) only have a small number of fault paths ($\eta = 49 : 35$, $\eta = 81 : 78$), which is counterintuitive. The instance $\eta = 49$ has a considerable runtime, yet exhibits only a few fault paths, while it is sort of the other way around for $\eta = 74$. Initially, this seems to contradict some crucial assumptions. Another noteworthy aspect is that the outlier in terms of the number of fault paths ($\eta = 74 : 2146$) also has the largest average fault path length ($l^a_{i_{74}} = 10.96$, cf. third plot), which is plausible. Yet, the second largest fault path length instance ($l^a_{i_{23}} = 10.19$) has an extremely lower number of fault paths ($\eta = 23 : 81$, cf. second plot). Hence, the prolonged runtimes observed can only be attributed to the number of fault path deviations (cf. Fig. 17). Accordingly, the runtime is not only determined by the number and length of ground truth fault paths, but also by the length and number of determined (incorrect) fault paths.

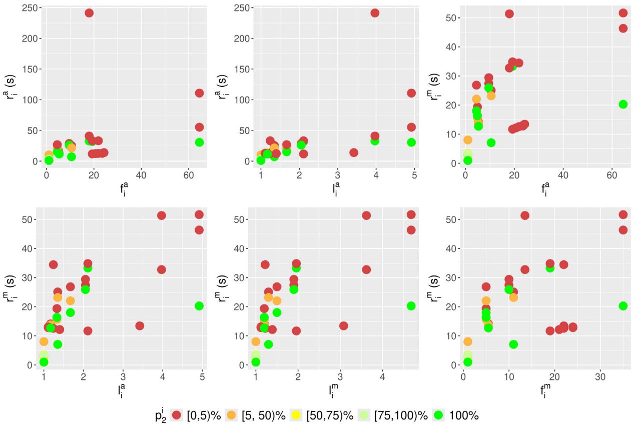


FIGURE 16. Runtime analysis.

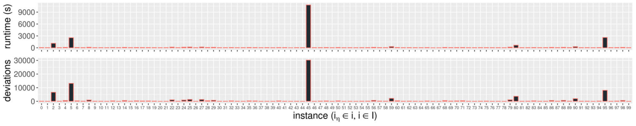


FIGURE 17. Comparing runtime and deviations.

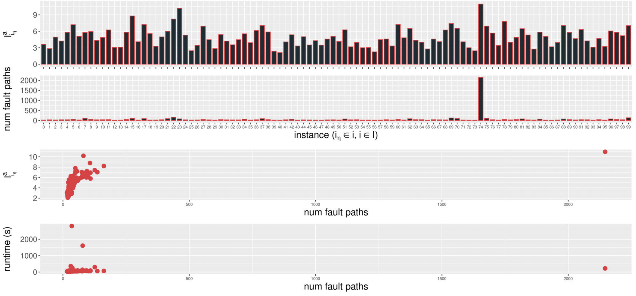


FIGURE 18. Outlier analysis.

Thus, it is not the actual number of fault paths (35) that is of consequence in the case of $\eta = 49$, but rather the significant fault path deviation of 9802. The rationale is that all potential paths must be generated and considered, regardless of their correctness. Again, the fault paths are generated a posteriori on the basis of the identified anomalies and the symbolic knowledge of causal links. Essentially, this is a post-hoc fault path generation based on graph traversal (depth-first). It thus appears reasonable to correlate the runtime with the sum of fault path deviations and the ground truth fault paths, as this should perfectly correlate with the runtime (cf. Fig. 19).

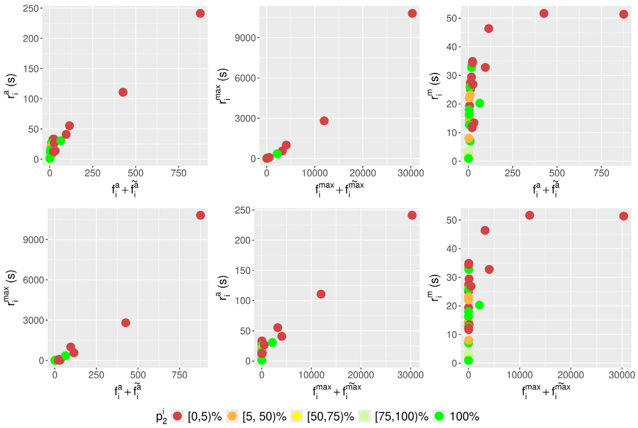


FIGURE 19. Runtime analysis.

The clearest correlation can be seen in plots two and four. Plot two correlates r_i^{max} with $f_i^{max} + \tilde{f}_i^{max}\ \forall i \in I$. This shows a perfect positive correlation ($\rho = 0.99, p \ll 0.001$), which confirms that the runtime is most dominantly determined by the number of fault paths, both ground truth ones and virtually found ones. The fourth plot correlates r_i^{max} with $f_i^a + \tilde{f}_i^a\ \forall i \in I$, again with a perfect positive correlation ($\rho = 0.98, p \ll 0.001$). The others show a similar picture, while, as expected, the average and median can wash out the effect somewhat due to large outliers. Consequently, the runtime of the fully automated part of the system is primarily determined by the post-hoc generation of fault paths. The system's efficiency in a real-world, step-by-step diagnostic scenario is primarily contingent upon the reaction times of humans that provide the input signals, i.e., the human interactions. In cases where all data is entered into the system instantaneously, this analysis holds true. Otherwise, the runtime is determined by human interaction times, which are of course subjective and situation-based. Model inference times are negligible in all experiments with UCR time series datasets from various domains, as well as for real-world recordings from the automotive domain in [20]. They can be considered instant for practical purposes, i.e., in context of the other elements of diagnosis.

$\max_{i \in I} r_i^a = 241.3\ s$ is by far the longest average runtime, resulting from $i\ := <\ 10\_20\_90\_95\ >$. It is of interest to examine its FPs (cf. Fig. 20). While it has only slightly more ($\bar{FP} = 7.3$) than $<\ 5\_20\_90\_95\ >$ ($\bar{FP} = 6.2$) and $<\ 20\_10\_90\_95\ >$ ($\bar{FP} = 6.6$), it has a significantly larger number of anomalies in one case, and substantially more connected anomalies in the other, which makes it more probable that the FPs result in long fault paths. Obviously, all three sets share the lowest considered model accuracy $\gamma \in [0.9, 0.95]$. As previously discussed, the total number of fault paths has the greatest impact on runtime. The number of fault paths in turn depends on the fault path lengths due to all the permutations and branches that result from them. This line of reasoning is perfectly aligned with the second plot of Fig. 34, which demonstrates a very high positive correlation of $\rho(f_i^a, l_i^a) = 0.81$ ($p \ll 0.001$). FNs are generally less likely as there are far fewer components that actually have an anomaly and can thus lead to FNs. Depending on $\alpha \in \{0.01, 0.05, 0.1, 0.2\}$ there are either 99%, 95%, 90% or 80% regular components, i. e., a massive class imbalance. Therefore, it is expected to have 4, 9, 19 or 99 times as many FPs as FNs. For instance, in the case of $\alpha = 0.2$ and $|C| = 129$, there are 26 anomalies and 103 regular components, i. e., ≈ 4 times as many regular components, and thus it is expected that there are also 4 times as many FPs as FNs, which is exactly what can be seen in plots two and four of Fig. 20 for the $\alpha = 0.2$ sets.

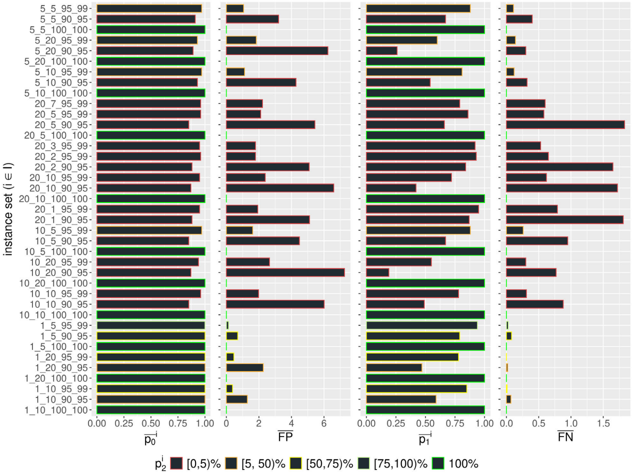


FIGURE 20. Performance analysis.

2) PERFORMANCE ANALYSIS

We must emphasize that only the case of $\gamma = 1.0$ really evaluates the correct functioning of the diagnosis system. The other cases are theoretic reflections on the nature of the problem as well as some guidance on when the system is still expected to meet the practitioner's expectations (evaluation under certain circumstances). As a first sanity check, all instance sets with $\gamma = 1.0$, i. e., $<\ \alpha\_\beta\_100\_100\ >$, achieve 100% ground truth matches (cf. Fig. 20), i. e., every ground truth fault path is found and no additional ones ($p_2^i = 100\%\ \forall i \in I_{<\alpha\_\beta\_100\_100>}$). In cases with $\gamma = 1.0$, all other configurations of the instances are irrelevant, the results will always be entirely correct. This evaluates the correct functionality of the system if everything is purely deterministic. The ground truth match percentage p_2^i is simply counting the boolean values for the instances $i_\eta \in i$, i. e., how many of them are solved entirely correctly. This emphasizes that even a single misclassification that is not compensable can cause a *False* here and thus have a huge impact on the result, it is an *all-or-nothing* metric. A similar argument can be made for the average ground truth fault path score $\bar{p_1}^i$, only somewhat more fine-grained. It is a stricter metric compared to $p_0^{i_\eta}$, since here not only the actual anomalies must be found, but also no regular components may be classified as such, i. e., FPs are harmful. Accordingly, the results are worse in certain configurations. Now to the practically more realistic scenario without completely perfect models, i. e., $\gamma < 1.0$. Here, the configuration, i. e., the nature of the problem instance, is very critical for the reliability of the approach. Crucially, there can be a $\bar{p_0}^i = \bar{p_3}^i = 1.0$ and still $p_2^i \neq 100\%$. Because of FPs, the system may find more anomalous links or anomalies than there actually are. First, we consider the less bright green colored instance sets in Fig. 20, i. e., $p_2^i \in [75, 100)\%$: Here, $\alpha = 0.01$, $\beta = 0.05$, and $\gamma \in [0.95, 0.99]$. In this case, on average, each anomalous link is also identified ($\bar{p_0}^i = 1.0$), but $\bar{p_3}^i = 0.97$ and only 94% of the ground truth fault paths on average ($\bar{p_1}^i = 0.94$). As can be seen in plots two and four in Fig. 20, this is due to very few FPs and FNs. Consequently, it can already be concluded that this practically realistic configuration leads to almost perfect results. Then to the yellow cases, i. e., $p_2^i \in [50, 75)\%$: Three instance sets, i. e., three configurations, ended up in the group: < 1_5_90_95 >, < 1_10_95_99 >, < 1_20_95_99 >. In all three cases, there are comparatively few anomalies (only 1 for $|C| = 129$). The first variant has inferior model accuracies, but also a less connected network of components, which leads to more isolated anomalies. In such a scenario, less accurate models are less damaging. The other two sets have more accurate models, but also more connected components. It looks as if the system still finds every anomalous link on average in each case, but there simply is no link that was not found, all (non-existent) links of the empty set were found. $\bar{p_1}^i$ is further reduced. i =< 1_5_90_95 > and j =< 1_20_95_99 > show very similar performance ($\bar{p_1}^i = 0.79$, $\bar{p_1}^j = 0.78$). One has less connectivity but also less accurate models and the other has the more accurate models but the highest level of connectivity, which seems to cancel out the advantage of improved model accuracy. In this case, k =< 1_10_95_99 > is the best solved configuration ($\bar{p_1}^k = 0.85$), it is sort of the compromise between the two previous ones. The model accuracy is improved, but the connectivity is not increased as much as in < 1_20_95_99 >. The problems are mainly due to FPs (cf. second plot in Fig. 20) that clutter the fault paths, noise in a sense. FNs could lead to missed anomalous links, which did not happen in this group. In fact, there are FNs, as can be seen in the fourth plot, but $\bar{p_0}^i = \bar{p_0}^j = \bar{p_0}^k = 1.0$. j has zero FNs on average, but i has 0.08 and k has 0.01 FNs on average. Thus, there are instances with 1 FN and yet $p_0^{i_\eta} = 1.0$, e. g., $\eta = 13$. The reason for this is that $p_0^{i_\eta}$ counts links; if there is only one anomaly, there are no links. As a reminder: $p_0^{i_\eta}$ is based on fault paths, i. e., it is about edges in identified fault paths. As introduced in (7), it considers the intersection of predicted ones and ground truth ones. This means, however, that fault paths containing only one component are disregarded in this metric. The rationale behind this is that it is insufficient to find all the anomalies without the correct connections. Also, in the case of the three instance sets, there cannot be more than one FN, as there is only one anomaly in each case. This leads to both $\hat{F}_{i_\eta}$ and $\hat{A}_{i_\eta}$ being empty. In such a case, $p_0^{i_\eta} = 1.0$ (cf. (7)). If there is no fault path that was not found, all (non-existent) fault paths of the empty set were found. However, the FNs lead to missed anomalies in the case of i and k: $\bar{p_3}^i = 0.92, \bar{p_3}^j = 1.0, \bar{p_3}^k = 0.99$. In the end, all three configurations are still solved rather well. Then to the more problematic group of $p_2^i \in [5, 50)\%$ with quite a number of configurations: < 5_5_95_99 >, < 5_20_95_99 >, < 5_10_95_99 >, < 10_5_95_99 >, < 1_10_90_95 >, < 1_20_90_95 >. Two of them still reach a perfect $\bar{p_0}^{<1\_10\_90\_95>} = \bar{p_0}^{<1\_20\_90\_95>} = 1.0$. However, these are again cases with $\alpha = 0.01$, i. e., only one anomaly and $\bar{p_3}^{<1\_10\_90\_95>} = 0.93, \bar{p_3}^{<1\_20\_90\_95>} = 0.98$. The worst is $\bar{p_0}^{<5\_20\_95\_99>} = 0.93$ with $\bar{p_3}^{<5\_20\_95\_99>} = 0.96$. The other three are somewhere in between. While these values still appear rather satisfying, there are heavy reductions in terms of $\bar{p_1}^i$, particularly in the case of i =< 1_20_90_95 >, which has dropped to $\bar{p_1}^i = 0.47$. This is particularly significant when comparing it to the only slightly different configuration < 1_5_90_95 > from the yellow group, compared to which we observe a reduction of 32% due to the increased β. Similarly poor results are obtained for i =< 5_20_95_99 >, which drops to $\bar{p_1}^i = 0.6$, and j =< 1_10_90_95 > ($\bar{p_1}^j = 0.59$). These two configurations are clearly not solved satisfactorily from a practical perspective. The other three are better, but probably still inadequate for practical purposes. The reasons for this are again mainly FPs, i. e., regular components treated as anomalies, but also some FNs. Finally, the last group with $p_2^i \in [0, 5)\%$ is far from practically reasonable. Unsurprisingly, there is no configuration in this group with $\bar{p_0}^i = 1.0$. Practically all configurations in this group are infeasible, the most extreme even go down to $\bar{p_1}^i = 0.19$, which is the case for i =< 10_20_90_95 >.

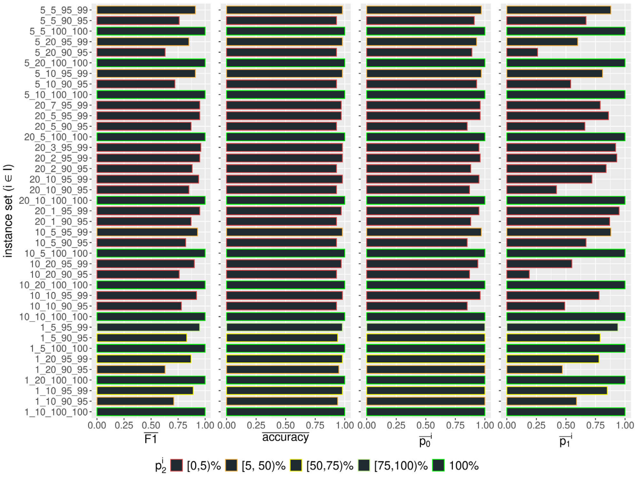


FIGURE 21. Classifications vs. Overall performance.

For such a high number of anomalies (α) and such a high connectivity (β), a model accuracy like this (γ) is completely insufficient. As you can see in plots two and four in Fig. 20, there are a lot of FPs in these configurations and also way more FNs compared to the previous configurations. The FNs are mainly due to the increase of α in conjunction with the low γ values. When there are no or very few anomalies, there is also little room for FNs, i. e., for missed anomalies. In these configurations, the potential for FNs is increased, so that more anomalies are missed.

Obviously, in the group of instance sets with $\gamma = 1.0$, all five metrics (four plots and ground truth match) displayed in Fig. 21 are also 1 or 100, respectively. In the end, it is also worth noting that $\bar{p_0}^i \geq 0.85 \, \forall \, i \in I$ (cf. Fig. 21), even in case of challenging configurations.

The accuracy $\frac{tp+tn}{tp+tn+fp+fn}$ clearly depends not only on γ, but also on the number of classifications (until convergence), i. e., not only on the uncertainty itself, but also on the number of uncertainties that are chained together, i. e., α and β. It essentially compares correct predictions with all predictions, which in turn represents the model accuracy if enough classifications are performed. In principle, high γ values lead to high accuracies, but with an increased $c_r^{i_\eta}$, incorrect classifications receive less weight, making it a decisive factor. Therefore, the accuracy deviates from γ with suboptimal models and very few classifications. With enough classifications, the accuracy value converges to the expected value γ. The actual number of positives is $n_p = \alpha|C|$ based on the known class distribution, so the expected number of TPs is $t_p = \gamma n_p$. This assumes that there are no or few entirely missed anomalies, i.e., that most of the n_p are actually classified. Reflections on misses and compensations can be found below. Likewise, $t_n = \gamma(1-\alpha)|C|$, $f_p = (1-\gamma)(1-\alpha)|C|$, and $f_n = (1-\gamma)\alpha|C|$. This leads to the expected accuracy in (11).

$$\lim_{n_c^{i_\eta} \to \infty} \frac{\gamma\alpha|C|+\gamma(1-\alpha)|C|}{|C|(\gamma\alpha+\gamma(1-\alpha)+(1-\gamma)(1-\alpha)+(1-\gamma)\alpha)} = \gamma \tag{11}$$

Obviously, this is all based on the expected values t_p, t_n, f_p, and f_n and only holds on the basis of the *law of large numbers* and thus a certain number of classifications $n_c^{i_\eta}$. How many depends on various characteristics, but looking at Fig. 21 suggests that $\bar{n_c}^i$ was sufficient for each instance set $i \in I$. Thus, the accuracy plot in Fig. 21 is rather uninteresting, it just resembles γ. The next metric to consider is the precision, i. e., $\frac{tp}{tp+fp}$:

$$\lim_{n_c^{i_\eta} \to \infty} \frac{\gamma\alpha|C|}{\gamma\alpha|C|+(1-\gamma)(1-\alpha)|C|} = \frac{\gamma\alpha}{\gamma\alpha+(1-\gamma)(1-\alpha)} \tag{12}$$

Thus, the precision converges to a term that depends on γ and α, i. e., it shows the relationship between the model's accuracy and the class distribution, indicating how the expected precision is affected by those factors. However, when $\alpha = 0.5$, i. e., when the classes are balanced, it again converges to γ. And finally the recall $\frac{tp}{tp+fn}$:

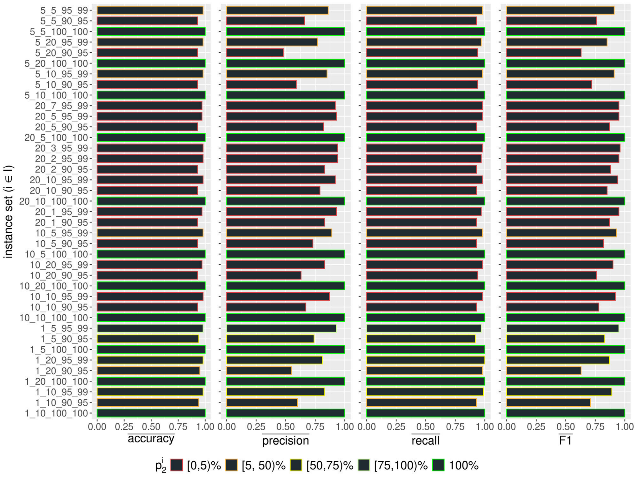


FIGURE 22. Evaluation metrics for classifications.

$$\lim_{n_c^{i_\eta} \to \infty} \frac{\gamma\alpha|C|}{\gamma\alpha|C|+(1-\gamma)\alpha|C|} = \gamma \tag{13}$$

The expected recall again converges to γ, just as the accuracy. It is essentially the accuracy within the positive class distribution. Since the $F1$ score balances precision and recall, it is clear what it depends on. In conclusion, these metrics do not have to be analyzed empirically, it is known what to expect analytically. It can only give insights w.r.t. the question of whether there already had been enough classifications for convergence for each instance set. Finally, precision is the most interesting metric since it is not necessarily converging to the model accuracy.

Fig. 22 shows that the recall is rather high ($\approx \gamma$) throughout the instance sets, i. e., the system is able to recall anomalies quite well, which is particularly significant for diagnosis systems, as FNs are so undesirable in diagnostic problems. Recall essentially measures how many relevant cases were identified. Missing even one true anomaly could have serious consequences in practical scenarios. As the second plot shows, the precision is not as high, so some regular components are classified as anomalies. The accuracy of the anomaly prediction is less critical because the most important aspect of diagnosis is to find the actual problems. This is reflected in the recall metric and also visualized by the previously considered $p_0^{i_\eta}$. If the system detects all anomalous links and only considers a few more, actually non-anomalous links, as anomalies, it is usually not as bad. However, it is important to note that reaching a higher recall than precision is not a result of some prioritizing mechanism built into the system, it is purely the result of the problem configuration. As explained, it is simply less likely to misclassify an anomaly compared to a regular signal, but this is just due to the fact that there are less anomalies to misclassify and not a result of the system being better at recognizing true anomalies. This is a quite crucial difference. If we would consider instance sets with anomaly percentages of, e. g., 50, this effect would no longer be present (precision would also be $\approx \gamma$). However, as argued before, this is neither reasonable from a practical perspective, nor does it make much sense to use the presented diagnosis system in such a scenario. The presented system is designed for problems where anomalies are the rare case, and having anomalies all over the place would also compromise the notion of regular versus anomaly. To give a specific example: In instance set $i =< 5\_20\_90\_95 >$, we see a very poor precision, but still a quite high recall. This is due to a large number of FPs, as you can see in Fig. 32. So the system identified most of the crucial issues, but the fault paths are cluttered with some additional components that actually do not show any anomalous behavior. The practical consequences of this highly depend on the application domain, but the plots allow to gain a good understanding of whether a particular configuration would be feasible in the considered domain.

3) MISSED ANOMALIES

$p_0^{i_\eta}, p_3^{i_\eta}$ and recall may seem to measure very similar aspects, they are not equal, though. First, we have to note that there is a difference between TP and n_p, i. e., between the number of true positive classifications and the actual ground truth number of positive instances. Likewise for TN and n_n. Some ground truth positive components may not be classified at all. Furthermore, even with zero FNs, the system does not necessarily find all anomalies and anomalous links due to the abortion criterion of stopping when arriving at a component classified as regular, i. e., not following *affected-by* relations of negatively classified components. This raises the question of missed classifications that are not counted as FNs, although they are ground truth positives. They are not counted at all, which leads to the interesting additional metric of how many ground truth anomalies are missed with the approach, either directly due to FNs, i. e., misclassified anomalies, or as a result of this due to the abortion criterion, or even based on the abortion criterion in the case of a TN. A simple case illustrating both indirect scenarios is shown in Fig. 23. If the first classification is a TN, the anomalies are missed due to the abortion criterion. If, on the other hand, it is a FN, the two anomalies are missed due to a false classification. Thus, some can be missed as a cost for early stopping (efficiency reasons), i. e., TNs but still some anomalies following it. This is the result of some *affected-by* relations that are not relevant in the case, the anomaly is not propagated further. This can happen as *affected-by* does not stand for guaranteed error propagation, it is just possible that an error is propagated from the root cause to the end of an *affected-by* chain. Yet this cannot happen in the synthetic cases: The instances are constructed in such a way that there is always at least one error code pointing to some ground truth fault path. Even if we assume the first component in Fig. 23 to be a distractor, which is the only case in which the scenario of a TN classification before anomalies is possible, there would be a ground truth fault path $\{C \rightarrow B\}$. So even if the corresponding input error points to e. g., $\{A, B\}$, it would be possible to find the anomalies via the suspect component B. This scenario is therefore not possible in the synthetic case. However, the case of missed anomalies due to FNs is just as possible as in real-world scenarios. FNs can also trigger the abortion criterion so that there could be a branch that is not investigated because of early stopping. If there are anomalies in this branch, they are not reflected in the $F1$ score. Those are not FNs, they were not classified at all. Hence, there can be an $F1$ score of 1.0, although many anomalies may be missed. It is simple to find a degenerate example where after one FN (or TN in real-world scenarios) there is an arbitrary high number of missed anomalies. Therefore, one misclassification can in principle lead to an arbitrary number of missed anomalous links due to a long chain of *affected-by* relations. We have no solution for the unfortunate case where an anomaly is hidden and not reachable via error code or chain of *affected-by* links. There is a difference between situations in which there is no path to the anomaly, and those in which there is a path, but the error is not propagated. The first (and second) case is not possible in the synthetic scenario as it is ensured to have such a path, but possible in practice in situations of an incompleteness of the underlying symbolic knowledge representation that can be resolved by extending it.

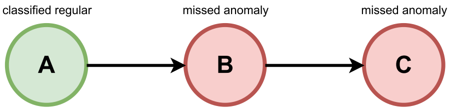


FIGURE 23. Illustration of missed anomalies.

A workaround could be to increase the probability of finding them, e. g., via an enhanced connectivity, although this is obviously not something that can be invented in a practical scenario, it is given based on the properties of the domain. But one can at least reason about likelihoods of such cases. It is more likely to miss something in a very sparse graph. The system has to follow some compromise between increasing the likelihood of finding an issue and still keeping the diagnosis effort limited, i. e., prevent doing an exhaustive search over all components. It is another argument to avoid FNs with a higher priority than FPs, as mentioned, the priority in diagnosis problems anyway. It is reinforced here, FPs lead to redundant classifications, longer fault paths, etc., but FPs are aspects we consider problematic that actually are not; it would be way worse to miss components that are indeed erroneous (FNs). In a nutshell: Both FPs and FNs should be avoided, but FPs are not as bad, they just result in being overly cautious in a way – it is more important to find problems than to not call anything a problem that is not. Nevertheless, it is also suboptimal to e. g., replace a wrongly identified root cause (FP) in [20] that is in fact functioning perfectly. Moreover, one should not understate the potential practical costs of FPs in terms of unnecessary recursive investigations along causal chains. Even if these chains are typically not extensive, depending on β and γ, it nonetheless entails additional effort. In case of an FP, either the entity of diagnosis has no problem at all, or there is another problem and another diagnosis run has to be started. Then it is very unlikely to have another misclassification at the same component. Obviously, this holds only when having a second step that verifies, a *human-in-the-loop*. If something hints that the system may be wrong, the human always has the option to not follow the suggestions, or at least to repeat the process. This should be highlighted: When there are results or recommendations that are not entirely convincing, then it is not implausible to simply repeat the process, because it is very unlikely to again have the same false prediction. It is non-deterministic in a way due to changes in the records. To conclude, the difference between recall and both $p_0^{i_\eta}$ and $p_3^{i_\eta}$ is that the latter measure how many anomalies or anomalous links of the ground truth were found, whereas recall is only considering the confusion matrix, so that missed anomalies are not considered at all. Hence, even with a perfect recall, there are not necessarily perfect $p_0^{i_\eta}$ and $p_3^{i_\eta}$ due to the abortion criterion. As anticipated, it can be of interest to measure the missed anomalies. The number of ground truth anomalies per instance is known, i. e., $|C|\alpha$, and also the number of found anomalies $fp + tp$. In most cases, the difference between the two is negative and in some cases it is zero, which means that the system usually finds more anomalies than expected due to FPs. Sometimes it finds the exact number of expected anomalies in the experiments, but never less than the expected. It is also interesting to consider the number of missed anomalies directly due to misclassifications (not as a consequence), i. e., $m_1^{i_\eta} := fn$. $\bar{m_1}^i \in [0.0, 1.9]$ across all the instance sets $i \in I$. Finally, we also measure the entirely missed ones, i. e., anomalies that were not even considered due to the abortion criterion (based on FN): $m_2^{i_\eta} := |C|\alpha - m_1^{i_\eta} - tp$. $\bar{m_2}^i \in [0.0, 0.3]$. In the end, all missed anomalies can be found with $m_3^{i_\eta} := |C|\alpha - tp$. Obviously, $m_1^{i_\eta} + m_2^{i_\eta} + tp = |C|\alpha$ and $m_1^{i_\eta} + m_2^{i_\eta} = m_3^{i_\eta}$. By this, we can tell that there are very few cases of entirely missed components, but there are some. The potential for this would be greater if there were longer fault paths, i. e., more chained anomalies. In our case, the anomalies are distributed rather uniformly across the component space; if they were more clustered, there would be more misses. This could be further analyzed in future work, as anomalies are not necessarily always evenly distributed. On the contrary, in practice it is probably often the case that anomalies are co-occurring in subsystems. The benchmark assumes a uniform distribution of anomalies across the component space, which may not accurately reflect all practical scenarios, especially those where faults are typically clustered within specific subsystems. However, in the context of industrial anomaly detection, both cases are observed: There are domains with uniformly distributed anomalies across the component space, in particular industrial settings with multiple parallel fault origins, but also domains with clustered anomalies in certain subsystems. There is evidence that fault detection and isolation should not be limited to local subsystems, but rather applied to large-scale systems as in this work, both with single and multiple fault origins [70]. As [71] emphasizes, a significant body of literature on fault diagnosis focuses on a single fault at a time, without addressing the diagnosis of root causes of multiple simultaneous faults. The authors argue that industrial systems are vulnerable to multiple faults concurrently, which may or may not be interconnected, thereby increasing computational complexity. It is noteworthy that while the evaluation in this work focuses on the rather uniform distribution of multiple parallel faults, it does include a certain degree of clustered anomalies. The clusters are only unlikely to be very large depending on the configuration.

Now we can explain a seemingly counterintuitive observation: There are quite high $\bar{F}1$ scores for instance sets $i \in I$ with very low p_2^i, e. g., the red-colored ones in Fig. 21. But $\bar{p_1}^i$ is still quite high in these cases, such that only a small fraction of the ground truth fault paths is not found. On the other hand, a lower $\bar{F}1$ score always has negative impact on the overall ground truth match performance. The correlation $\rho(\bar{F}1, p_2^i)$ is only 0.56 ($p \ll 0.001$), but this is due to the fact that even a perfect $\bar{F}1$ does not guarantee a high p_2^i (cf. Fig. 23). The situation of lower $\bar{F}1$ will always lead to worse p_2^i, though. The $\bar{F}1$ score does not correlate with $\bar{n_c}^i$ ($\rho = 0.05, p \gg 0.5$, cf. Fig. 24), which confirms that $\bar{n_c}^i$ is sufficient for the precision and recall to converge in case of each instance set $i \in I$. However, even few misclassifications can have a huge impact in terms of ground truth matches. Finally, the $\bar{F}1$ score fairly perfectly correlates with $\bar{p_1}^i$ ($\rho = 0.86, p \ll 0.001$, cf. Fig. 24). This shows that a high ratio of found ground truth fault paths is insufficient to end up with a high p_2^i and also that quite few anomalies are missed (not classified due to the phenomena illustrated in Fig. 23) on average. The $\bar{F}1$ score does not correlate as strongly with $\bar{p_0}^i$ ($\rho = 0.49, p \ll 0.05$, cf. Fig. 24), which again is a result of the above reasoning. There is a higher correlation with $\bar{p_3}^i$ ($\rho = 0.72, p \ll 0.001$), e. g., due to cases of fault paths of length one, but also not as high for the same reasons. Nevertheless, a higher $F1$ score is generally helpful for $p_0^{i_\eta}$ and $p_3^{i_\eta}$. Thus, the above counterintuitive observation is partially due to missed anomalies, i. e., anomalous components that are not wrongly classified and thus part of $F1$, but that are not classified at all due to the reasons illustrated in Fig. 23. Again, $F1$ and $p_0^{i_\eta}$ and $p_3^{i_\eta}$ are correlating as one would intuitively expect, just not as much due to the discussed reasons. Intuitively, when $F1$ increases, i. e., when the classifications improve, the end results also improve, but this is only a necessary and not a sufficient condition (except for $\gamma = 1.0$). Even with almost perfect classifications, one does not necessarily arrive at equally good end results; this also

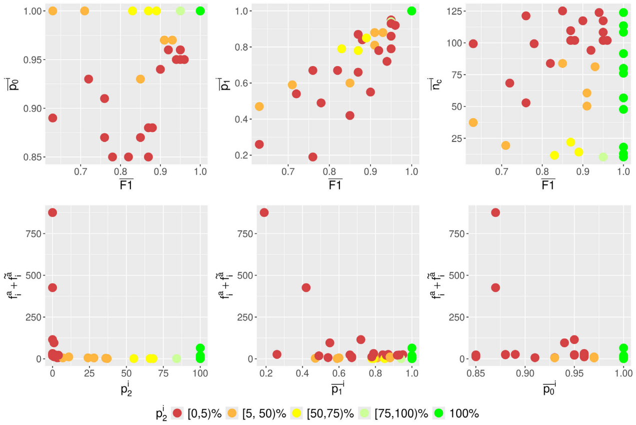


FIGURE 24. ***F*1 analysis.**

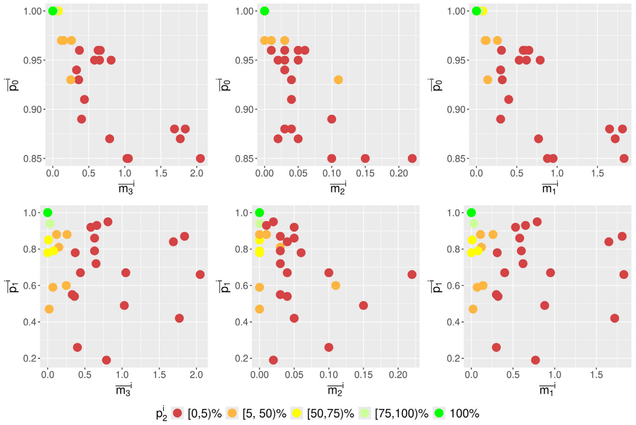


FIGURE 25. Impact of missed anomalies.

depends on the symbolic side, e. g., due to the termination criterion.

The next aspect to analyze is the impact of missed anomalies on the overall performance of the system shown in Fig. 25. We clearly see the expected negative correlations. The three columns within the $\bar{p_0}^i$ row show almost the same level of correlation. The row for $\bar{p_0}^i$ exhibits a way stronger significant correlation ($\rho \approx -0.85, p \ll 0.001$) compared to the row for $\bar{p_1}^i$, in which plots one and three roughly agree ($\rho \approx -0.35, p \ll 0.05$). The middle plot shows a stronger negative correlation $\rho(\bar{p_1}^i, \bar{m_2}^i) = -0.48, p \ll 0.05$. This is to be expected because $p_0^{i\eta}$ is not affected by FPs and thus purely determined by FNs and early stoppings, whereas the fault paths are. Interestingly, the correlations between $\bar{p_3}^i$ and the $\bar{m_n}^i$ are somewhat between the two rows in Fig. 25 ($\rho \approx -0.65, p \ll 0.001$). Since most of the misses in the considered scenarios are caused by direct misclassifications, this is unexpected. The misses due to FNs on isolated components have no influence on $\bar{p_0}^i$, but on $\bar{p_3}^i$. The stronger correlation of misses with $\bar{p_0}^i$ compared to $\bar{p_3}^i$ can be explained as follows. Generally, the entries are similar to the $\bar{p_0}^i$ rows, but there is an additional vertical line of entries at $\bar{m_n}^i$ that is very close to zero. A fairly low $\bar{p_3}^i$ with almost zero $\bar{m_n}^i$ can occur, as one miss leads to an anomaly score of zero for instances with only one anomaly. Nevertheless, this is very unlikely and still leads to $\bar{p_3}^i \geq 0.92$ for the considered model accuracies over 100 instances of a set. Ultimately, it depends on α. If $\alpha = 0.01$, one miss has a high impact on the performance, if $\alpha = 0.2$, one miss is not so decisive, which is why the worst $\bar{p_3}^i \approx 0.92$ covers the entire $\bar{m_3}^i$ range [0.0, 2.1]. For $\bar{p_0}^i$, it simply takes more misses to really have an impact.

4) PERFORMANCE IMPACT OF PARAMETERS

Based on the previously discussed convergence, we can directly look at the correlation between the average model accuracy γ (instead of the accuracy score) and the most important overall performance metrics (cf. Fig. 26).

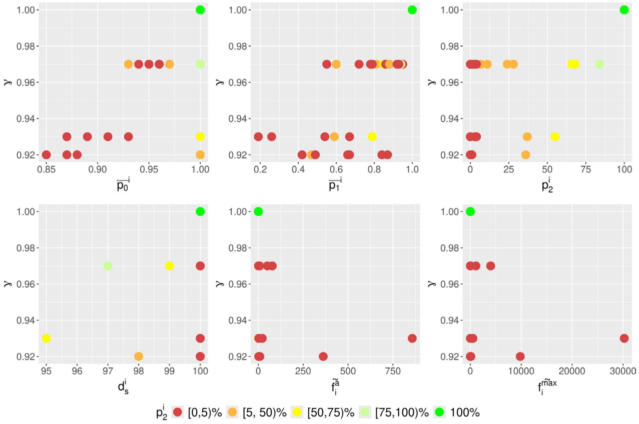


FIGURE 26. ***γ* against performance.**

There is a clear and obvious tendency that higher γ values appear with higher $\bar{p_0}^i$ ($\rho(\gamma, \bar{p_0}^i) = 0.76, p \ll 0.001$) and higher $\bar{p_1}^i$ ($\rho(\gamma, \bar{p_1}^i) = 0.76, p \ll 0.001$) values. But the model accuracy is certainly not the only factor determining the overall performance (if $\gamma < 1.0$). This is clearly illustrated by the instance sets with $\gamma = 0.97$, where there are $p_1^{i\eta}$ values ranging from below 0.6 to above 0.9. Likewise for $p_0^{i\eta}$ and $p_3^{i\eta}$, although the ranges are a lot smaller with [0.93, 1.0] and [0.96, 1.0]. Additionally, higher γ values appear with higher p_2^i values ($\rho(\gamma, p_2^i) = 0.74, p \ll 0.001$), which is intuitively plausible. The *diagnosis success percentage* d_s^i, which measures the cases in which the system ends up with a diagnosis, i. e., at least one fault path, is not significantly correlating with γ, $\rho(\gamma, d_s^i) = 0.28, p \gg 0.05$, and seems to be mostly determined by other factors. Likewise for the $\tilde{f}_i^a$ and $\tilde{f}_i^{max}$, which show insignificant negative correlations. Intuitively, better models lead to less deviations. Similarly, Fig. 27 shows a tendency that lower β values seem to be beneficial, but it is not in isolation able to explain the results and also way less obvious. There is no significant correlation with $\bar{p_0}^i$ and $\bar{p_3}^i$, but quite some negative correlation with $\bar{p_1}^i$

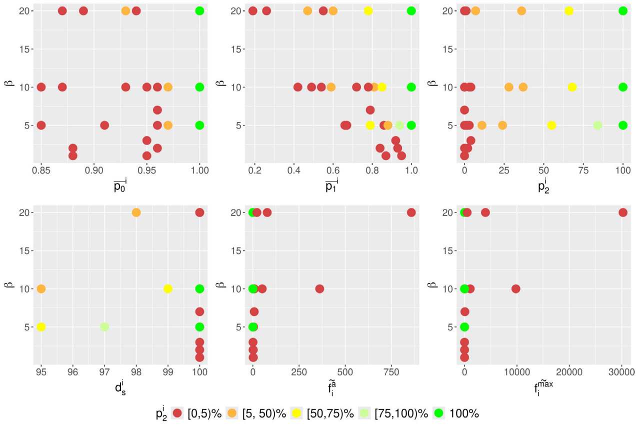


FIGURE 27. β **against performance.**

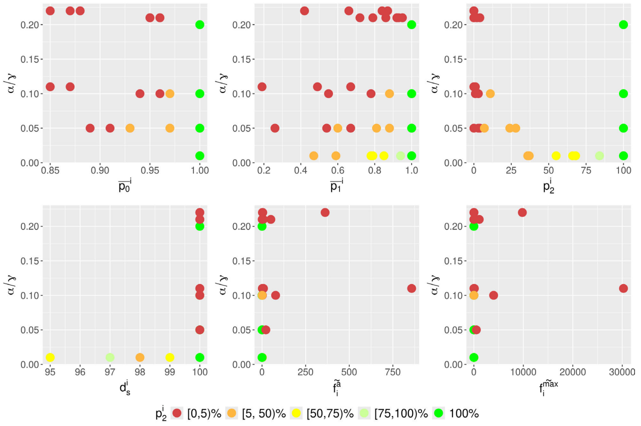


FIGURE 28. $\frac{\alpha}{\gamma}$ **against performance.**

($\rho(\beta, \bar{p_1}^i) = -0.41, p \ll 0.05$). The remaining ones again do not show a significant correlation. In summary, we see a sort of cluster in the bottom-right quadrant of the second plot, i. e., lower *affected-by* scores more often occur with higher $p_1^{i_\eta}$. Overall, it seems as if increasing β does more damage than it helps. Although this notion is a bit misleading: β is part of the domain knowledge and not some parameter that can be arbitrarily lowered or increased. A higher β simply represents a more complex problem that is thus harder to solve with high accuracy. The tendency for worse solutions is thus simply the result of an increased problem complexity in combination with an insufficient model accuracy. Therefore, the positive and negative aspects of it do not seem to balance out in the parameter space considered in this paper, the negative impact (increased complexity) seems to be stronger. However, the increased connectivity is required for compensation of misclassifications, which is analyzed in Fig. 39.

Again, considering the ratio $\frac{\alpha}{\gamma}$ (cf. Fig. 28), we can observe the cluster for smaller ratios, i. e., more accurate models or less anomalies seem to help for an overall better performance, which is not surprising. With $\rho(\frac{\alpha}{\gamma}, \bar{p_0}^i) = -0.45, p \ll 0.05$, $\rho(\frac{\alpha}{\gamma}, p_2^i) = -0.45, p \ll 0.05$ and $\rho(\frac{\alpha}{\gamma}, d_s^i) = 0.4, p \ll 0.05$, there are only three significant correlations with a magnitude $\geq$ 0.4. Interestingly, for the first two metrics, a smaller ratio leads to better performance, but for d_s^i this trend is inverted. However, this is only due to the instance sets with $\alpha = 0.01$. In other words: An increase in α helps to avoid situations in which only one misclassification leads to *no_diag*. Furthermore, it is surprising that it does not correlate significantly with $\bar{p_3}^i$, which is closely related to $\bar{p_0}^i$. The resulting ranges are also comparable: $\bar{p_3}^i \in [0.9, 1.0]\,\forall i \in I$ and $\bar{p_0}^i \in [0.85, 1.0]\,\forall i \in I$. To answer this question, it is reasonable to consider α and γ in isolation against $\bar{p_3}^i$. There is no significant correlation between $\bar{p_3}^i$ and α. However, $\rho(\gamma, \bar{p_3}^i) = 0.92$ ($p \ll 0.001$). Thus, α is irrelevant for the anomaly score, because in principle there can be $\bar{p_3}^i \in [0.0, 1.0]$ for each α, depending on γ. This is different for $\bar{p_0}^i$ for the reasons already mentioned. Each value of α enables the entire observed $\bar{p_3}^i$ range [0.92, 1.0], which is not the case for $\bar{p_0}^i$ due to $\alpha = 0.01$, where $\bar{p_0}^i = 1.0\,\forall i \in I_{<1\_\beta\_\gamma^{LB}\_\gamma^{UB}>}$.

α by itself is not able to explain the overall performance either. Obviously, fewer anomalies seem to help, and the same three variable pairs exhibit the largest correlations, i.e., $\rho(\alpha, \bar{p_0}^i) = -0.42$ ($p \ll 0.05$), $\rho(\alpha, p_2^i) = -0.43$ ($p \ll 0.05$) and $\rho(\alpha, d_s^i) = 0.4$ ($p \ll 0.05$). We also considered the ratio $\frac{\gamma}{\beta}$. There is not a single significant correlation between $\frac{\gamma}{\beta}$ and the six metrics considered in the previous Figures. Generally, the worse the model, the better the graph should be connected in order to compensate wrong classifications. However, this also has the negative side-effect of leading to more classifications and thus also to more potential for misclassifications. As discussed before, $p_0^{i_\eta}$ is not affected by this as much as $p_1^{i_\eta}$.

Fig. 29 also has a quite clear message: The quotient $\frac{\bar{n_c}^i}{\gamma}$ should be small for good performance. There is a strong negative correlation $\rho(\frac{\bar{n_c}^i}{\gamma}, \bar{p_0}^i) = -0.58$ ($p \ll 0.001$). Intuitively, when there are more classifications or a weaker model accuracy, there is a lower $p_0^{i_\eta}$, except for $\gamma = 1.0$, then $n_c^{i_\eta}$ is irrelevant. Again, there is no significant correlation with $\bar{p_3}^i$, as suboptimal results can be obtained with very few (e. g., one) classifications. Furthermore, there is a quite high negative correlation $\rho(\frac{\bar{n_c}^i}{\gamma}, p_2^i) = -0.52$ ($p \ll 0.001$) with the same intuitive understanding. Finally, there is a positive correlation of the same magnitude with d_s^i. The other plots show insignificant correlations. In case of $\bar{p_0}^i$ this is due to the perfectly accurate models, the trend is clearly visible in the plot. For the last two plots, i. e., the fault path deviation, this is due to the large outliers and the cases with perfectly accurate models. Trivially, the performance is worse when having too many classifications with a too poor model accuracy.

$n_c^{i_\eta}$ is by itself not as well-suited to explain the effect. γ is always very close to 1.0, which means that the denominator seems neglectable. Showing the same tendency, the correlations are a bit weaker, though, e. g., $\rho(\bar{n_c}^i, \bar{p_0}^i) =$

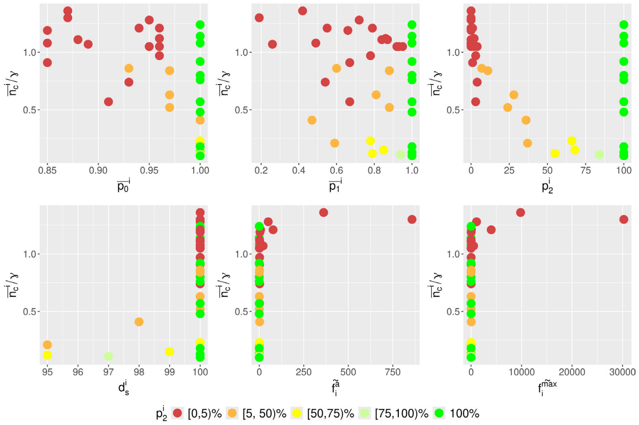


FIGURE 29. $\frac{n_c}{\gamma}$ against performance.

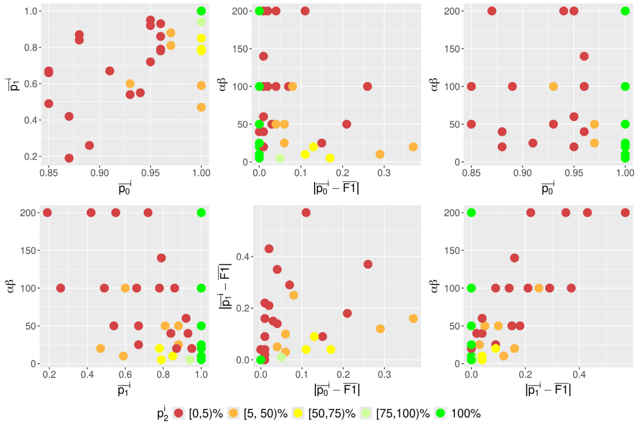


FIGURE 30. $\alpha \cdot \beta$ against performance.

-0.53 ($p \ll 0.001$) and $\rho(\bar{n_c}^i, p_2^i) = -0.48$ ($p \ll 0.05$). This is due to the cases of perfectly accurate models.

Now we start some reflections on the differences $|\bar{p_0}^i - \bar{F}1|$ and $|\bar{p_1}^i - \bar{F}1|$. First, we consider a large $|\bar{p_0}^i - \bar{F}1|$: When $\bar{F}1 \ll \bar{p_0}^i$, there are wrong classifications that did not affect $p_0^{i_\eta}$ as much as they did $F1$, e. g., a lot of FPs that did not prevent the further search for truly anomalous links. As argued before, in some situations it can even help to explore some unexplored areas in the network. On the other hand, $\bar{F}1 \gg \bar{p_0}^i$ can happen in situation of early stopping, e. g., there could be only one FN that is not affecting $F1$ that much that hides a long chain of missed anomalies behind it, causing a poor $p_0^{i_\eta}$. Then to the case of a large $|\bar{p_1}^i - \bar{F}1|$: One could assume that $F1$ should generally be closer to $p_1^{i_\eta}$, because both directly depend on classification results, i. e., the confusion matrix. The confusion matrix is based on the uncertainty of the models, but also on $n_c^{i_\eta}$, which is in turn based on α and β. A large difference between the two is not surprising, though, because misclassifications are more damaging to $p_1^{i_\eta}$. A single misclassification can affect a whole lot of fault paths. The first plot in Fig. 30 shows a strong correlation $\rho(\bar{p_0}^i, \bar{p_1}^i) = 0.64$ ($p \ll 0.001$) between the two most important performance metrics, which is intuitively plausible as they both depend on accurate models and a reasonable problem structure. The next three plots consider the product $\alpha\beta$ on the y-axis, which may be better suited to show their influence than the ratio, since it enables to also judge the magnitude. The second plot does not show a significant correlation $\rho(\alpha\beta, |\bar{p_0}^i - \bar{F}1|) = -0.24$ ($p \gg 0.05$), not even if $\gamma = 1.0$ instances are filtered out. Nevertheless, this is an indication that with smaller products, there are more early stoppings or FPs. The likelihood of FPs increases with smaller α values, since there are less ground truth anomalies. The following plots show similar or slightly larger correlation coefficients, except the final one. The last plot shows a strong positive correlation $\rho(\alpha\beta, |\bar{p_1}^i - \bar{F}1|) = 0.57$ ($p \ll 0.001$). An interpretation could be that with increased β, the number of fault paths that are affected by a misclassification increases. Overall, there is a tendency of better performance for smaller products. In particular, the fourth plot clearly shows a cluster of good results in the bottom-right quadrant.

Fig. 31 shows the same plots for the ratio $\frac{\alpha}{\beta}$ instead of the product. The second plot shows a stronger negative correlation compared to the product. However, it only becomes significant when filtering out the $\gamma = 1.0$ cases, then it is $\rho(\frac{\alpha}{\beta}, |\bar{p_0}^i - \bar{F}1|) = -0.45$ ($p \ll 0.05$). This is an indication that with smaller ratios, i. e., larger connectivity due to β, there are more early stoppings or FPs, e. g., due to more classifications. In particular when the ratio is < 1, i. e., when the connectivity is higher than the anomaly percentage, $\bar{p_0}^i$ is usually way better than $\bar{F}1$. Again, $F1$ is only concerned with the performed classifications, whereas $p_0^{i_\eta}$ also includes components that have not even been classified. If $p_0^{i_\eta}$ is better than the $F1$ score (the average $\bar{F}1$ across all instance sets is 0.9, the average $\bar{p_0}^i$ is 0.96, and the average $\bar{p_3}^i$ is 0.97), it can be due to FPs. FPs negatively affect the $F1$ score, but have no negative effect on the $p_0^{i_\eta}$ score, they can even lead to an accidental discovery of the previously described case of a hidden anomaly. The third plot shows a slightly stronger negative correlation ($\rho(\frac{\alpha}{\beta}, \bar{p_0}^i) = -0.29$) compared to the product. However, it is narrowly insignificant ($p = 0.07$). Nevertheless, in this case a larger β, i. e., a smaller $\frac{\alpha}{\beta}$ ratio, seems beneficial for $p_0^{i_\eta}$, which indicates the compensational effect of β. As argued before, increasing β has negative side-effects for the overall performance, but those do not affect $p_0^{i_\eta}$. $p_0^{i_\eta}$ benefits from the increased $c_r^{i_\eta}$. In summary, β in isolation does not exhibit a significant correlation with $\bar{p_0}^i$ (cf. Fig. 27), since it is mostly affected by other factors. However, in relation to α, larger β values improve the $p_0^{i_\eta}$ performance. Crucially, the final plot again only shows a significant correlation when filtering out the $\gamma = 1.0$ cases ($\rho(\frac{\alpha}{\beta}, |\bar{p_1}^i - \bar{F}1|) = -0.4, p \ll 0.05$), which is the one with a very strong positive correlation in case of the product (cf. Fig. 30). Hence, the sign is different, because this time larger β lead to smaller ratios, which in turn confirms the

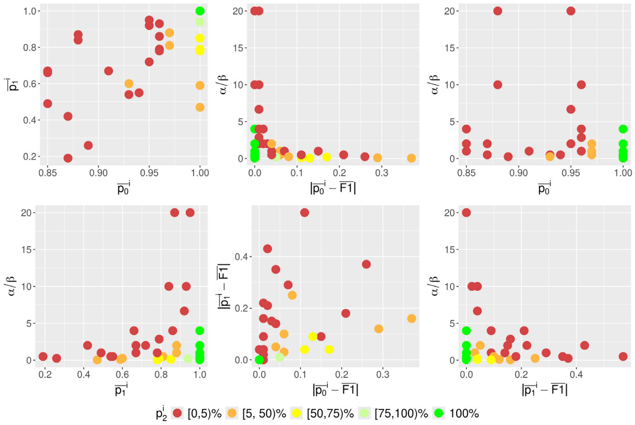


FIGURE 31. $\frac{\alpha}{\beta}$ **against performance.**

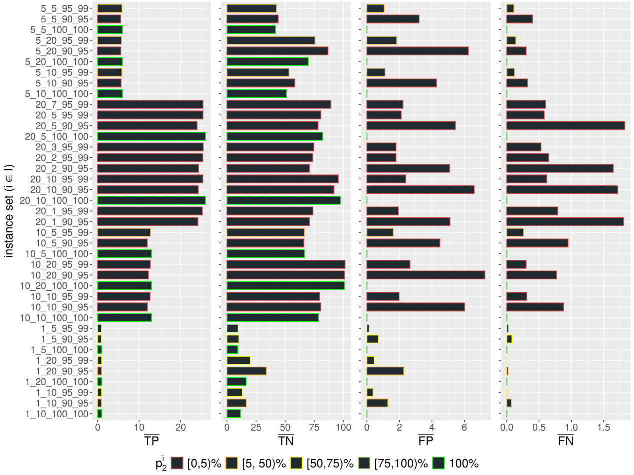


FIGURE 32. **Confusion matrix.**

interpretation that the number of affected fault paths increases with larger β. The key observation is that the difference is usually negative, meaning that the average $F1$ is usually better than $\bar{p_1}^i$. $p_0^{i_\eta}$ had the advantage of not being affected by FPs, the fault paths are, though. Very crucially even, a single FP can turn a whole set of otherwise correctly identified ground truth fault paths into a mismatch.

Fig. 32 shows the confusion matrix plots. FPs and FNs were already discussed as part of Fig. 20. It is expected to have 4, 9, 19 or 99 times as many FPs as FNs, depending on α. The same is of course true for the TNs compared to the TPs. Obviously, the number of TPs is closely related to α, it naturally increases with α. One can clearly observe four groups along the instance sets in the average TP plot with only very little variance within each group. This grouping perfectly resonates with $\alpha \in \{0.05, 0.2, 0.1, 0.01\}$. For the TNs, there is no such clear grouping, which might be surprising at first. It seems that the system is a lot more reliable in predicting anomalies compared to regular signals. FPs can lead to more classifications, while FNs can prevent them. For FNs, the effect is less visible, because there simply are less FNs. The effect can be approximated by subtracting the additional TN classifications due to FPs from the TNs; $FP \cdot \beta \cdot |C|$ should be a reasonable approximation. However, it also depends on the number of components already classified at the time of the considered FP. It is insufficient to simply subtract the FPs, since each FP can increase $c_r^{i_\eta}$ and thus lead to more TNs. One would have to subtract all the TN classifications they led to based on β. The TPs are a lot less disturbed, because there simply is not much alternative. Either the true anomaly is predicted as one, then it is a TP, or the true anomaly is predicted as negative, which leads to a FN. But the potential for this is rather small, as mentioned before, the potential for the other way around is 4 to 99 times as high. Thus, the system is better at predicting positives, because there simply is less potential for error. Generally, it is rather unlikely to classify something incorrectly, since $\gamma_i \geq 0.9 \, \forall \, i \in I$, but if it happens, it is a lot less likely to predict wrong and miss an anomaly with this. Simply because there are fewer anomalies than regular signals.

5) FAULT PATH ANALYSIS

The second plot in Fig. 33 shows that there are massive outliers in terms of the number of fault paths. $\max_{i \in I} f_i^a \approx 65$, while $\max_{i \in I} f_i^{max} = 2146$. Note the logarithmic x-axis. The combinatorial explosion again happens in the cases of $< 20\_10\_\gamma^{LB}\_\gamma^{UB} >$. The fault path length depends on the combination of α and β. This can be seen very well in plot six of Fig. 34. In general, more fault paths and longer fault paths, which usually coincide, are harder to match completely correctly, which is why p_2^i is way worse for larger numbers of fault paths that are longer, except when having perfectly accurate models. Essentially, more ground truth fault paths result from longer fault paths, i. e., fault paths involving more components that are interconnected, leading to more permutations.

Fig. 33 is best explained by the following Fig. 34. First, there is the insignificant $\rho(\beta, f_i^a) = -0.16$ ($p \gg 0.05$) (cf. first plot). This is because β only has an influence in relation to α. The second plot shows the expected almost perfect correlation of $\rho(f_i^a, l_i^a) = 0.81$ ($p \ll 0.001$). This is because longer fault paths are based on more anomalies, which in turn allow more permutations and thus increase the number. The third and fourth plots indicate that α is a good predictor for f_i^a and l_i^a, i. e., $\rho(\alpha, f_i^a) = 0.76$ ($p \ll 0.001$) and $\rho(\alpha, l_i^a) = 0.5$ ($p \ll 0.05$). The most crucial point is that the product $\alpha\beta$ correlates perfectly well with the number of fault paths and their lengths, i. e., $\rho(\alpha\beta, f_i^a) = 0.69$ ($p \ll 0.001$) and $\rho(\alpha\beta, l_i^a) = 0.96$ ($p \ll 0.001$). However, the effect on f_i^a is way stronger for α, which is a precondition for any fault paths. β, on the other hand, is not able to affect f_i^a by itself, only in combination with α, i. e., β controls it when α is high enough. Many anomalies are not chained together, but mostly isolated. Nevertheless, the β influence on the length of fault paths and thus also on the number of fault paths is absolutely decisive and confirmed by the aforementioned correlation

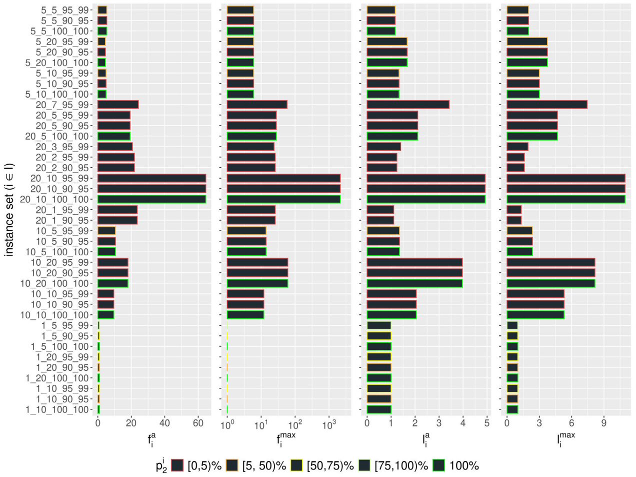


FIGURE 33. Fault path analysis.

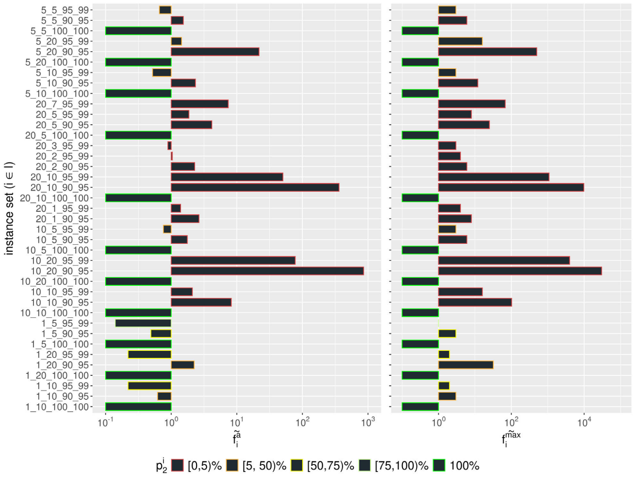


FIGURE 35. Fault path deviation.

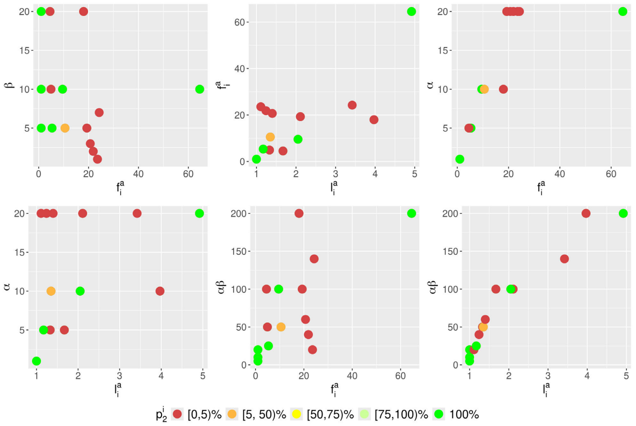


FIGURE 34. Fault path correlations.

coefficients, in particular by the increase in the correlation coefficient with l_i^a from 0.5 to 0.96 by multiplying α and β.

Finally, we can estimate the actual number of feasible permutations, i. e., f_i^a with the earlier introduced formula (2). The average approximation percentage across all instance sets is 89.53, i. e., the actual number of fault paths is on average 89.53% of the estimated. The percentage range across all instance sets is [58, 150], the median is 87.83%. Thus, the approximation marginally overestimates, but is very useful to judge the magnitude, at least for the parameter ranges considered in this work. Also of interest are the fault path deviations shown in Fig. 35. The range of fault path deviations within one instance set is rather large, [2, 30246] in the most extreme case. Note the logarithmic x-axis. So, if it is important not only to find all anomalies, but also not to call anything an anomaly that is not, the results can get arbitrarily poor in some configurations. Each instance set has at least one instance with 0 fault path deviations, except $<$ 10_20_90_95 $>$ with at least 2. Usually, $\tilde{f}_i^{max} < 105$, in many cases even $\tilde{f}_i^{max} < 10$, but with $\tilde{f}_i^{max} \in \{3991, 30246, 499, 9802, 1085\}$ for $i \in \{<$ 10_20_95_99 $>$, $<$ 10_20_90_95 $>$, $<$ 5_20_90_95 $>$, $<$ 20_10_90_95 $>$, $<$ 20_10_95_99 $>\}$ there are some extreme exceptions. Again, each set with $\gamma = 1.0$ achieves 0 deviations according to expectation.

Unsurprisingly, $\tilde{f}_i^a$ is highest in instance sets with the longest and most fault paths with imperfect models. Although not always: $i =<$ 5_20_90_95 $>$ seems to not fit that scheme. It has only as many and as long fault paths as many other sets with smaller $\tilde{f}_i^a$, but in this case the huge number of deviations can be explained with the comparatively low γ. There are some more instance sets with such a level of γ, but those either have fewer anomalies, are less connected, or are exploding in terms of deviations, e. g., $i =<$ 10_20_90_95 $>$. The even more extreme case of $\alpha = \beta = 0.2$ turned out to be infeasible due to excessively large f_i^a. In fact, some more nuanced tests revealed that the combinatorial explosion already occasionally occurs for instances with $\alpha = 0.2, \beta = 0.12$. Also, all instance sets with a significantly large $\tilde{f}_i^a$ belong to the red category in terms of p_2^i, i. e., $p_2^i \in [0, 5)$%. This is expected, when there are large deviations, not a lot of ground truth fault paths are matched. $i =<$ 10_20_90_95 $>$ has the largest $\tilde{f}_i^a$. Compared to $j =<$ 10_20_95_99 $>$ it has less accurate models, which explains the difference. Compared to the second highest $\tilde{f}_k^a$ for $k =<$ 20_10_90_95 $>$, it has equally accurate models and less anomalies that are more connected. Once again, the connectivity seems to be very damaging: i has less anomalies, which should be easier to solve, but the increased connectivity leads to a higher potential for misclassifications. The beneficial effect of a higher connectivity, which is that a missed anomaly could be reached again via another path, seems to be canceled out. The misclassified component in such a case is not considered again, but components that it could be affected by that were not considered could be reached and potentially identified as anomalies then. It is of interest to measure how often this happens: Anomalous affecting components that were missed due to a misclassification and then found via another path. Therefore, anomalies that are not recognized as such, which triggers the abortion criterion for the considered branch. This means that potential anomalies within this branch are missed, except when they are reachable via another path, which would exactly be the compensating factor one intuitively might assign to β. Intuitively, one could assume that the higher β, the better it should be in case of inaccuracy. This is due to the fact that at some point each misclassification is reached again. Obviously, each component is only classified once. This is a problem when the wrongly classified component is either the root cause, or the only connection to a potentially long branch of anomalies, i. e., the critical point. Otherwise it might be possible to reach it via another path. As a future extension, it might be reasonable to search for such critical paths in the causal network and treat them separately, e. g., by classifying certain components more than once if they appear on a critical path. It is a lot less likely to misclassify a component twice. There is generally a tradeoff: The number of classifications (and corresponding recordings) should be small to minimize the effort and runtime, but at the same time there should be as few mistakes as possible. Of course, there will also be the vice versa case: A previously correct classification followed by a misclassification. It might be reasonable to approach this heuristically: If there is an anomaly that turns out to be confirmed by other causal links on the critical path, it is assumed to be true, otherwise it could be verified again.

While the system may appear very fragile to a single misclassification on a critical path, it is rarely a matter of concern. First, FPs are not problematic in this regard, as they may lead to additional work, but not to any missed anomalies. FNs, on the other hand, are very rare on their own and even less frequently leading to additionally missed anomalies. We analyze this effect in detail in the following section and show that missed anomalies due to FNs are extremely rare.

6) COMPENSATION OF POTENTIALLY MISSED ANOMALIES

We are interested, in particular, in how well the system is able to compensate model inaccuracies through structural knowledge, i. e., whether there is some clear compensation and thus a beneficial effect, or whether it is more balanced out and β has positive and negative effects on the overall performance. The positive effect is that it can compensate FNs by reaching potentially missed anomalies again via another link. The negative effect is that it leads to higher $c_r^{i_\eta}$ and thus also to more potential for misclassifications. The question is whether these two effects cancel each other out. First, we introduce the *compensation by affected-by savior* metric $c_1^{i_\eta}$ that is best described by Fig. 36.

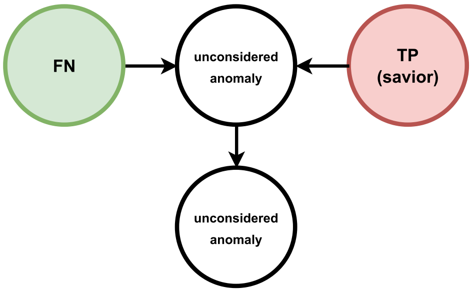


FIGURE 36. $c_1^{i_\eta}$ compensation.

Initially, we go through the *affected-by* relations of each FN and search for ground truth anomalies. If they were found via another link, there is a $c_1^{i_\eta}$ compensation (cf. Fig. 36). The following classification is not required to be correct for this, we consider the additional chance to find it as the compensation. If this is not the case, i. e., the ground truth anomaly is not found via another link, we check if it would have been possible via another link, this would be a *missed chance*, metric $c_2^{i_\eta}$, i. e., an unclassified component that has a link pointing to the missed one. Again, it is not about the classification result, and also not about whether the component is somehow reachable, we only count the existence of such a link. If both are not the case, it is counted as *no second chance*, metric $c_3^{i_\eta}$. Crucially, the *savior* is only counted once for one component, even when multiple FNs are pointing to the same potentially missed anomaly. Thus, it represents an exists relation. In addition, each component is counted at most once as a savior, so that the ground truth number of anomalies represents a natural upper bound for the number of saviors in an instance. The same holds for $c_2^{i_\eta}$. A missed chance is only counted once, even if there are many links that could have been used. It is worth noting that we do not follow the branches after a missed anomaly that could potentially include arbitrary many more missed anomalies. Fig. 37 is suited to confirm whether an increasing β is beneficial for the results. Surprisingly, in plot 1 in Fig. 37, it can be seen that $\bar{c_1}^i$ is not significantly correlating with β, i. e., $\rho(\beta, \bar{c_1}^i) = -0.15(p \gg 0.05)$. This indicates that the lowest considered $\beta = 0.01$ may still be sufficient to allow for the required compensation, or at least that not the entire β range is necessary, i. e., that a further increase in β beyond a certain point is no longer useful in terms of compensation. The parameter may essentially only vary in a range in which the lower fraction already enables the full amount of compensation, which renders the remaining larger part of the range $\beta \in [0.01, 0.2]$ irrelevant in this regard. Furthermore, in cases where anomalies are missed due to FNs there seems to also be always enough connectivity for compensation. And, on the other hand, when there are lower β and thus not enough connectivity for compensation, it is also very unlikely to miss anomalies, so that no $c_1^{i_\eta}$ is required. This is confirmed by the following correlation. If *saviors* would not reduce missed anomalies, there would be a high correlation between the two. There is a perfect correlation of $\rho = 1.0(p \ll 0.001)$ between $\bar{c_1}^i$ and potentially missed anomalies, i. e., anomalies that would have been missed without a *savior*. Thus, if there are a lot of missed anomalies, there is also a high $c_1^{i_\eta}$. If there are few missed anomalies,

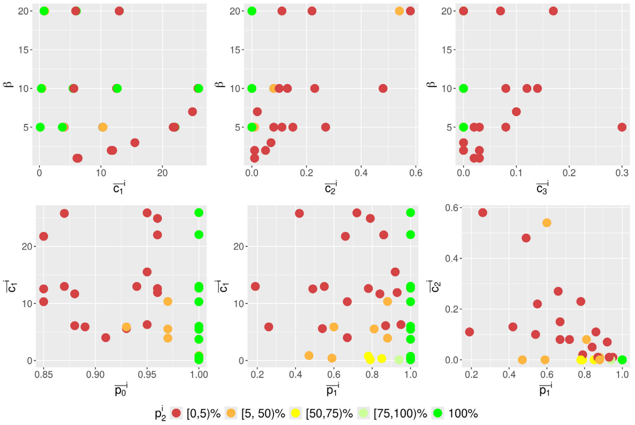


FIGURE 37. Compensation correlation.

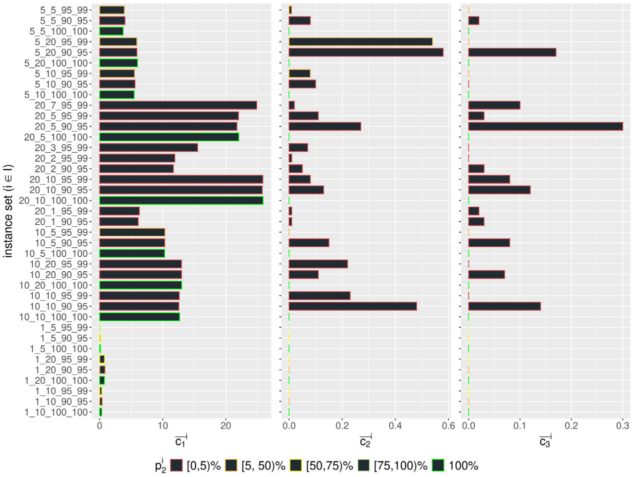


FIGURE 38. Compensation metrics.

there is a low $c_1^{i_\eta}$. There is a natural balance between the two: The system will not have a high $c_2^{i_\eta}$ without a high $c_1^{i_\eta}$, and vice versa. Because in such a scenario, the anomalies are sort of isolated, which is also not increasing $c_3^{i_\eta}$ as there simply are very few or no such missed anomalies.

We have already seen that high β values can have a quite negative impact on the overall performance (cf. Fig. 27), which shows that there might be an incentive to use the system in domains with lower connectivity. Although not too low either, because the system depends on a certain level of connectivity. However, this is only due to the fact that a higher connectivity is generally more complex and thus also more error prone. In conclusion, there is no tradeoff between good compensation and no huge negative effects. As seen, the compensation naturally scales with the missed anomalies and is always appropriate. The overall negative impact of a high β is not only a side effect of many classifications with imperfect models, but also a result of higher connectivity leading to greater impact on many fault paths. Again, this is only assessing the performance of the system under certain conditions, β is not something that can be arbitrarily lowered or increased as it reflects causal relationships between components of some real-world domain. The compensation in the case of the entire instance sets (cf. Fig. 37) is heavily based on the numbers of FNs (the potential for missed anomalies), i. e., $\rho(\bar{c_1}^i, fn_i) = 0.44$ $(p \ll 0.05)$. This is also confirmed by the very low numbers of missed chances displayed in the second plot of Fig. 38. If there are few missed chances in all scenarios, then there is also little room for compensation via increased β, indicating that β is already enough for many scenarios. $c_2^{i_\eta}$ is heavily based on β. Missed chances are more likely with a higher β because then it is more likely that there is another link, but at the same time they are less likely because then they are less likely to be missed due to the overall increased connectivity. This was confirmed by introducing the instance set with $\beta = 0.01$. Apparently, β positively correlates with the number of missed chances, i. e., $\rho(\beta, \bar{c_2}^i) = 0.45$ $(p \ll 0.05)$, but again only if the instances with $\gamma = 1.0$ are filtered out. As anticipated, there are few missed chances, but a high connectivity, i. e., a high β, clearly benefits it. Situations with no second chance are so rare across all considered instance sets (cf. Fig. 38), that it does not enable a further analysis. This is again for the same reason: $c_3^{i_\eta}$ scenarios appear when there are missed anomalies, but no further links, which happens almost never. This is a confirmation of the previous argument of a natural balance. As discussed, plots four and five in Fig. 37 are considering $\bar{c_1}^i$, which in this case only correlates with the FNs, not with β. The final plot is interesting, though. There are low $\bar{c_2}^i$ values, but it shows that missed chances have a huge impact on the overall performance. There is a clear negative correlation of $\rho(\bar{c_2}^i, \bar{p_1}^i) = -0.62(p \ll 0.001)$. An increasing $c_2^{i_\eta}$ usually goes hand in hand with a worse performance, which is intuitively evident.

To confirm the previous arguments, we now consider some low β instances in isolation. For these instance sets, there should be a clear correlation. We consider $< 20\_1\_90\_95 >$, $< 20\_2\_90\_95 >$, and $< 20\_5\_90\_95 >$, i. e., $\alpha = 0.2$, $\beta \in \{0.01, 0.02, 0.05\}$, $\gamma^{LB} = 0.90$, and $\gamma^{UB} = 0.95$. As expected, there is a very clear correlation of $\rho(\beta, \bar{c_1}^i) = 0.99$ between connectivity and compensation. The same holds for $\rho(\beta, \bar{c_2}^i) = 1.0$, and also for $\rho(\beta, \bar{c_3}^i) = 0.97$. Although these correlations are all insignificant due to the small sample size, they do provide an indication. Finally, there is also a significant perfect negative correlation $\rho(\bar{c_2}^i, \bar{p_1}^i) = -1.0(p \ll 0.05)$. Thus, the correlation between β and $\bar{c_1}^i$ may indeed end with $\beta \approx 0.05$. For higher values, there seem to be no significant changes. However, it is also crucial to note that the three instance sets considered here to analyze the β influence all share the maximum value of $\alpha = 0.2$, because this is the most interesting case for analysis. If there are few anomalies, then there is also little room for missed chances, etc. It is also interesting to consider the absolute value ranges over the entire instance sets. On average, $\bar{c_1}^i \in [0.13, 25.9]$, $\bar{c_2}^i \in [0.0, 0.58]$,

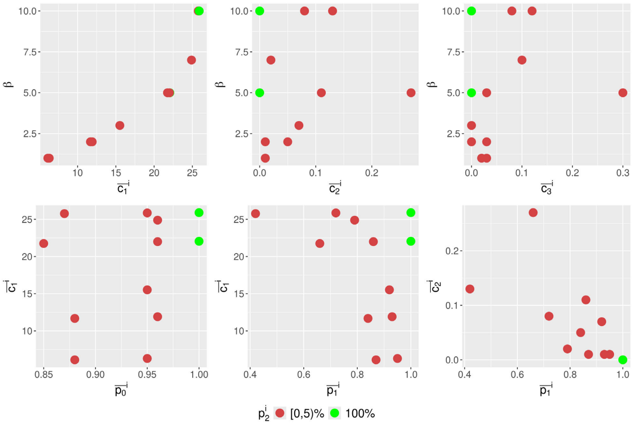


FIGURE 39. Compensation correlation ($\alpha = 0.2$).

and $\bar{c_3}^i \in [0.0, 0.3]$. For the three selected low β instance sets there are the following progressions with increasing $\beta \in \{0.01, 0.02, 0.05\}$. $\bar{c_1}^i$: 6.11 $\rightarrow$ 11.68 $\rightarrow$ 21.75, $\bar{c_2}^i$: 0.01 $\rightarrow$ 0.05 $\rightarrow$ 0.27, and $\bar{c_3}^i$: 0.03 $\rightarrow$ 0.03 $\rightarrow$ 0.3. Fig. 39 shows all solved instance sets with $\alpha = 0.2$, which are 12.

For those instance sets, we also see the expected clear correlation $\rho(\beta, \bar{c_1}^i) = 0.93$ ($p \ll 0.001$). Not for $\bar{c_2}^i$, though, $\rho(\beta, \bar{c_2}^i) = 0.4 (p \gg 0.05)$, which is a bit misleading in the above case of the three selected instances. The same lack of correlation can be seen in $\rho(\beta, \bar{c_3}^i) = 0.26$ ($p \gg 0.05$). We already provided an argument explaining both. Obviously, $\rho(\bar{c_2}^i, \bar{p_1}^i) = -0.67$ ($p \ll 0.05$) again shows a clear negative correlation. Therefore, in the end, the effect is mainly due to α. If α is sufficiently large, β correlates heavily with $c_1^{i_\eta}$.

The positive compensatory effect of β is clearly visible and it grows with the otherwise missed anomalies themselves, naturally balancing them. Nevertheless, it is of interest to determine whether the positive compensation effect or the negative impact grows faster with increasing β. For this we only consider the six instance sets with $\alpha = 0.2$ and $\gamma \in [0.95, 0.99]$ to analyze the isolated impact of an increasing $\beta \in [0.01, 0.1]$. Table 2 shows the results. The average number of misclassifications, i. e., $FN + FP$, increases, but not substantially, not even by one misclassification on average. Also, there is no influence on $\bar{p_0}^i$, $\bar{p_3}^i$ and p_2^i. $p_0^{i_\eta}$ and $p_3^{i_\eta}$ are again unaffected due to their resilience to FPs, and p_2^i would generally be affected, but is very low or 0 anyway for the instance sets with $\alpha = 0.2$ and this level of γ. One might expect a positive compensatory effect on $\bar{p_0}^i$, if there were misses before, it should get better. First, it could be canceled out: If there are about as many misses as compensations, the effect on $\bar{p_0}^i$ is invisible. However, this is not the case, as can be seen in Table 2, there are far more compensations than misses in every case. The misses remain relatively stable across all β, but $\bar{c_1}^i$ increases. Most of the misses are misclassifications, the rest is more or less negligible, so that the number of misclassifications does not change significantly. Moreover, an incorrectly classified component is not reclassified, so there is no way to compensate for this. Compensation is only useful for the unclassified anomalies behind such a misclassification, which are, however, very few. This is why $\bar{p_0}^i$ does not change, the compensations measure the potential to mitigate the effects of misclassifications, but not the misclassifications themselves. Since the effects of misclassifications are very small either way, there is almost no effect on $\bar{p_0}^i$. Thus, we see that the compensation potential is steadily increasing, although it is not really required for the instances considered so far, since the majority of the missed anomalies are immediate misclassifications. It would therefore be of interest to generate example instances with anomalies that are more clustered and not so evenly distributed across the entire component space.

TABLE 2. Analyzing the influence of increasing β.

β	$\overline{FN}+\overline{FP}$	$\overline{p_0}^i$	$\overline{p_1}^i$	p_2^i	$\overline{c_1}^i$	$\overline{m_1}^i$	$\overline{m_2}^i$	$\overline{m_3}^i$
0.01	2.7	0.95	0.95	0	6.3	0.79	0.02	0.81
0.02	2.4	0.96	0.93	2	11.9	0.65	0.01	0.66
0.03	2.3	0.95	0.92	4	15.5	0.53	0.05	0.58
0.05	2.7	0.96	0.86	2	22.0	0.58	0.05	0.63
0.07	2.8	0.96	0.79	0	24.9	0.6	0.03	0.63
0.10	3.0	0.95	0.72	0	25.9	0.62	0.03	0.65

Crucially, there is a major impact on $p_1^{i_\eta}$, which decreases massively with increasing β, and also a huge influence on $c_1^{i_\eta}$, which increases with β. The number of misclassifications thus increases more or less continuously by around 10% when going from $\beta = 0.01$ to $\beta = 0.1$. On the other hand, $\bar{c_1}^i$ grows continuously by around 311%. Hence, in this range, the growth of compensation is way stronger than the growth of misclassifications. However, it is not only the misclassifications themselves that are harmful, but also the number of fault paths that are affected due to the increased connectivity of the network. Therefore, we see $\bar{p_1}^i$ continuously decreasing from 0.95 to 0.72. From this we can tell that the compensation grows faster / stronger than the negative effects, but the negative effects are still significant, particularly with regard to $\bar{p_1}^i$. Consequently, the problem is not so much that there are more misclassifications, but that a misclassification has a stronger impact on the overall performance (p_2^i).

Fig. 40 shows the corresponding correlations for the six instances. Finally, the previously considered $\bar{m_2}^i$ shows that there are very little entirely missed anomalies. Thus, it indeed helps to have a more connected anomaly graph in order to compensate misclassifications that in turn also raise based on an increased connectivity. Intuitively, one could assume that it is all about the ratio $\frac{\alpha}{\beta}$, if this gets small, it should be beneficial. We saw the clustering of favorable results in the bottom-right quadrants in plots three and four in Fig. 31, i. e., a tendency of better results for smaller $\frac{\alpha}{\beta}$ ratios. Generally, large β should help with imperfect models and a certain amount of α, since there are many classifications

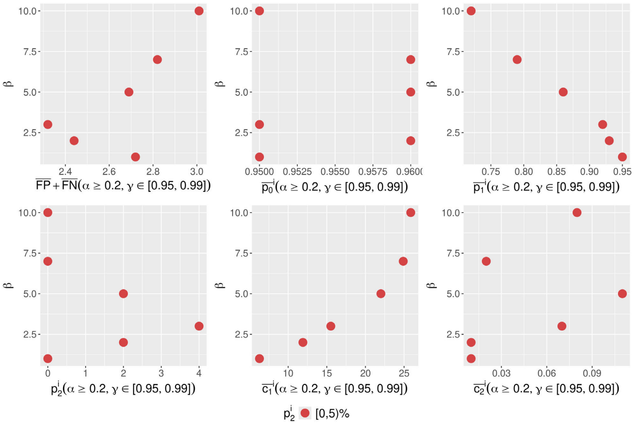


FIGURE 40. Analyzing the influence of increasing β.

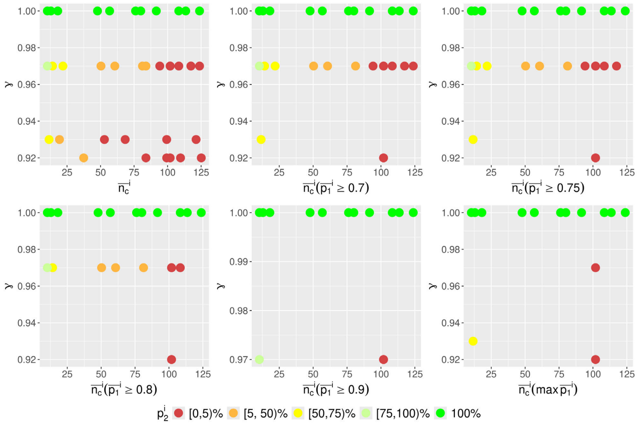


FIGURE 41. γ vs. $\overline{n_c}^i$ (filtered based on $\overline{p_1}^i$).

and thus also much potential for misclassification. These misclassifications should be better compensated when having a higher connectivity, i. e., a higher β. However, as we learned, the higher compensatory capability goes hand in hand with an increased potential for further misclassification and a more severe impact of misclassifications. In a way, there is always as much compensation as required in terms of missed anomalies, but negatively affected fault paths remain a concern.

7) BALANCING $n_c^{i_\eta}$ AND γ

Fig. 41 can answer the question of how many classifications $n_c^{i_\eta}$ can be reasonably performed at most based on a certain γ. However, as argued before, $n_c^{i_\eta}$ is based on α and β, so that these are implicitly part of the plot. Therefore, the statement could be rephrased as how many and how connected the anomalies should reasonably be, and how accurate the models must be in each case. It is essentially about the number of chained classifications versus the model accuracy. The first plot is already very insightful. If $p_2^i \geq 50\%$ is expected, it is advised to have $n_c^{i_\eta} < 25$, except for perfectly accurate models. The following four plots filter the $\bar{n_c}^i$ displayed on the x-axis based on increasing thresholds $t \in \{0.7, 0.75, 0.8, 0.9\}$ for $\bar{p_1}^i$. The results are not surprising. The fifth plot provides a good summary: If very good results are expected, i. e., $\bar{p_1}^i \geq 0.9$, it is advised to perform as few classifications as possible, or to obtain close to perfectly accurate models. The final plot is also somewhat interesting: There, we filter the x-axis based on the maximum $\bar{p_1}^i$ per γ value. $\gamma = 0.92$ has $\bar{n_c}^i > 100$, but with $p_2^i < 5\%$. In the previous sections, we have already seen how large $n_c^{i_\eta}$ values come about (α, β). In this section, we are only interested in the question of when the performance gets infeasible based on γ.

Fig. 42 considers the same question, but filters the x-axis based on $\bar{p_0}^i$ instead of $\bar{p_1}^i$. As argued before, $p_0^{i_\eta}$ is in a sense more critical as it directly measures the identified problematic links. $p_1^{i_\eta}$ can, for instance, be low due to a number of FPs, which is not that critical if all ground truth TPs are actually found. In plots two to four it can be seen that even with the worst $\gamma = 0.92$, it is possible to go up to $\bar{n_c}^i > 125$, i. e., $\bar{c_r}^i \approx 1$, if a $\bar{p_0}^i$ of 0.8 suffices. Plot five shows that even if $\bar{p_0}^i \geq 0.9$ is expected, $\gamma \geq 0.97$ is sufficient to go up to $\bar{c_r}^i = 1$ with $|C| = 129$. The final plot is filtered based on the maximum $\bar{p_0}^i$ per γ and shows that it is still reasonable to only perform as many classifications as absolutely needed, at best $\bar{n_c}^i < 25$ when $\gamma \in [0.93, 0.97]$. This shows that FNs do play a role as well.

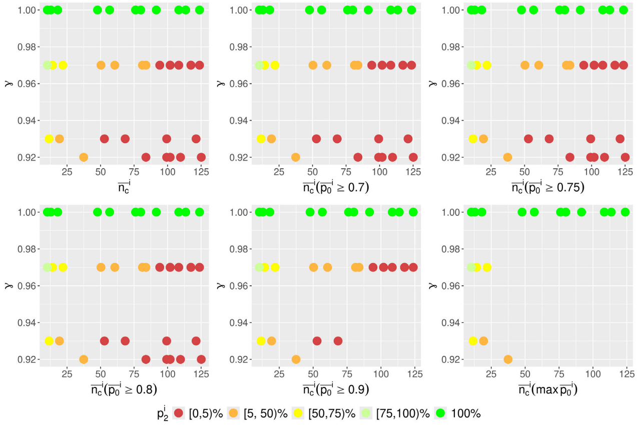


FIGURE 42. γ vs. $\overline{n_c}^i$ (filtered based on $\overline{p_0}^i$).

Fig. 43 again considers the same question, but filters the x-axis based on p_2^i, which is even stricter than $\bar{p_1}^i$. Plots two to four show the exact same results, where it is obvious that $\gamma = 0.97$ is the only imperfect model that is performing well enough to match the expectation, and only with an average $\bar{n_c}^i = 10.32$. If we ask for $p_2^i \geq 90\%$, only the perfectly accurate models are able to achieve this. Finally, this is another confirmation that it is beneficial to minimize $n_c^{i_\eta}$.

One final metric to be considered is the *diagnosis success percentage* d_s^i per instance set $i \in I$, which measures the fraction of instances that were solved with a diagnosis, i. e., at least one fault path. This is not measuring the quality of fault paths, it is only measuring whether there is some diagnosis at

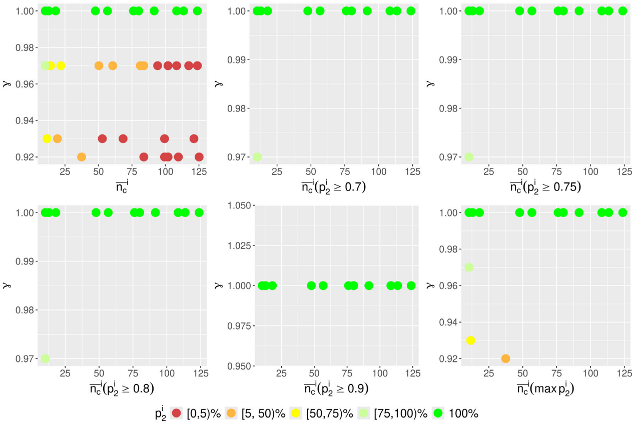


FIGURE 43. γ vs. $\overline{n_c}^i$ (filtered based on p_2^i).

all, i. e., an outcome that is not *no_diag*. As we only consider instances that actually comprise problems, this should be a rare occurrence. Most instance sets have $d_s^i = 100\%$, only < 1_10_90_95 >, < 1_10_95_99 >, < 1_20_90_95 >< 1_5_90_95 >, and < 1_5_95_99 > deviate, still all having $d_s^i \geq 95\%$. It does not surprise that all of these share $\alpha = 0.01$, as this should go hand in hand with very few classifications and anomalies. Because this can only happen when not a single anomaly is found, i. e., all input error codes lead to no anomalies, which is unlikely if there are many anomalies and fairly accurate models. In the $\alpha = 0.01$ cases, there is only one anomaly for $|C| = 129$, which means that a single misclassification can lead to *no_diag*.

C. SCALABILITY ANALYSIS

While the broad, systematic evaluation focused on scenarios with a fixed $|C| := 129$ due to the previously discussed reasons, the scalability in terms of performance and computational costs to cases with significantly larger or smaller numbers of components is demonstrated in the following. For this purpose, we consider additional instance sets:

- < 20_10_20_50_50_90_95_42 >
- < 50_10_20_50_50_90_95_42 >
- < 80_10_20_50_50_90_95_42 >
- < 250_2_5_50_50_90_95_42 >
- < 500_2_5_50_50_90_95_42 >
- < 1000_1_3_50_50_90_95_42 >

The majority of the sets are once again comprised of 100 instances. Due to the increased computational costs, the sets with $|C| = 500$ and $|C| = 1000$ are based on 10 instances each. Moreover, in such cases, smaller α and β values are used for the same reason, still providing practically meaningful examples. The fact that all considered instances are solved demonstrates the system's scalability to significantly larger and smaller scenarios. It is noteworthy, though, that an equivalent iterative version of the recursive causal sub-graph construction as well as fault path candidate generation had to be implemented in order to avoid exceeding Python's default maximum recursion depth in the cases of $|C| = 500$ and $|C| = 1000$. Without investigating all subtleties, each considered set led to $\bar{p_0}^i > 0.8$, which would be even higher if γ^{LB} and γ^{UB} were higher. Therefore, each additional set is successfully solved. It is evident that the most substantial change when increasing $|C|$ is the increase in r_i^a. The following progression is observed as the magnitude of C increases from $|C| = 20$ to $|C| = 1000$: $2.5s \rightarrow 16.8s \rightarrow 46.7s \rightarrow 125.2s \rightarrow 1329.9s \rightarrow 6583.8s$. $r_i^{max} = 16940.4s$ is extremely high for i =< 1000_1_3_50_50_90_95_42 >. Accordingly, the order of magnitude of $|C| = 1000$ may be considered a reasonable upper bound for many practical scenarios. To conclude, there are expected increases in runtime and memory consumption; however, the system still functions adequately on consumer hardware at least up to $|C| = 1000$ for the considered α and β.

Occasionally, experiments with smaller $|C|$ led to recursion depth issues as well before implementing an iterative version, e.g., i =< 250_10_20_50_50_90_95_42 >. In cases of $|C| = 250$ or higher, the runtime and memory consumption rapidly increases and may become intractable with high α and β values. Nevertheless, as long as it remains tractable, it may be feasible in practice to have increased runtimes to solve a hard problem. Moreover, assuming 10% anomalies is typically way beyond the number of parallel anomalies to be expected in most practical domains. While 250 components may appear manageable, having such a high degree of connectivity and such a high number of anomalies quickly leads to a combinatorial explosion in terms of possible anomalous paths. The complexity escalates rapidly. The greater the number of components involved and the higher the connectivity, the less anomalies should be expected to handle requirements in terms of runtime and memory. Prior to the implementation of the iterative version of the recursive processes, the system's scalability practically ended with $|C| = 250$. The bottleneck was the state *ISOLATE_PROBLEM_CHECK_EFFECTIVE_RADIUS* in Fig. 8. There, the maximum recursion depth was reached for larger C. Converting the algorithm to an equivalent stack-based iterative version improved the scalability in terms of $|C|$ as demonstrated by the above experiments with $|C| = 1000$.

To further assess the system's limitations, we examine one instance of < 1000_5_10_50_50_90_95_42 > and < 500_5_10_50_50_90_95_42 >. The system no longer crashes due to the maximum recursion depth, but due to the combinatorial explosion in the fault path computation, it is intractable for practical purposes and was terminated after 48 hours of runtime in both cases. The depth-first search-based fault path computation is operating with an anomaly graph comprising 157 edges and 25 nodes in the second case, which is intractable. To provide context, the most complex instance set in the main experiments of this work is $|C| = 129$ with $\alpha = 0.2$, i.e., 25 components, and $\beta = 0.1$. Thus, there are cases in which 25 components work with

$\beta = 0.1$ and a runtime of only a few seconds. Clearly, the connectivity is not absolute, but related to the number of components, i.e., there are a lot more connections with $|C| \geq 500$ and thus more edges. To verify, a sample instance from $<$ 129_20_10_50_10_100_100_42 $>$ has 23 edges, as expected, way less edges than 157. Consequently, it is reasonable to find a smaller upper bound. For this purpose, we consider $<$ 500_5_5_50_50_90_95_42 $>$. A sample instance of this configuration was solved with a runtime of $2707.5s$ and 19 fault paths. This particular instance is suitable to refer to a special case in the fault path approximation formula 2. The l_i^a approximation is negative, which leads to an empty sum in the f_i^a approximation, so that only $\lfloor \alpha |C| \rfloor$ is counted, which leads to a rather good approximation of 25. This occurs when $(\lfloor \alpha |C| \rfloor - 1)\frac{\beta}{2} < 1$ in 1, resulting in a negative logarithm. Due to the resulting empty sum in 2, the approximation continues to work. Finally, we consider the same configuration with $|C| = 1000$: $<$ 1000_5_5_50_50_90_95_42 $>$. With 933 edges, this is another instance that is clearly intractable. To summarize, for $|C| = 1000$, $\alpha = 0.01, \beta = 0.03$ worked, while $\alpha = 0.05, \beta = 0.05$ did not work. It is worthwhile to test the in-between scenario of $\alpha = 0.03, \beta = 0.03$, i.e., $<$ 1000_3_3_50_50_90_95_42 $>$. This instance is solved with a substantial runtime of $12506s$. With this, the scalability of the system is sufficiently explored.

D. CONCLUSION OF THE EVALUATION

The purpose of the evaluation was to draw conclusions and provide guidance with regard to the applicability of the presented diagnosis system in a wide variety of domains. In particular, one aim was to examine limitations. We evaluated the system by solving 4000 domain-agnostic problem instances and analyzed the results. The idea was to systematically test the neuro-symbolic diagnosis framework independently of the previous real-world use case in [20], as it is too restricted to draw general conclusions. First, we determined a reasonable approximation for the expected order of magnitude of the number of feasible fault path permutations based on partially connected anomaly graphs ($\beta \neq 1.0$). The average approximation percentage across all instance sets is 89.53, i. e., the actual number of fault paths is on average 89.53% of the estimated. The percentage range across all instance sets is [58, 150], the median is 87.83%. Thus, the approximation marginally overestimates, but is very useful to judge the magnitude. It was demonstrated that it is feasible to only consider $\alpha, \beta \in [0.0, 0.2]$ since $\bar{c_r}^i \approx 1.0$. $c_r^{i_\eta}$ obviously depends on the combination of α and the connectivity β. Naturally, it is advantageous to minimize $c_r^{i_\eta}$ or to pay close attention to obtaining very accurate models, i. e., to maximize γ, in particular with increased problem complexity through f_i^a and l_i^a. FPs lead to unnecessary additional classifications, while FNs can reduce $c_r^{i_\eta}$ (by circumventing the *affected-by* links through incorrectly triggering the termination condition). While the termination criterion may appear very fragile to a single misclassification on a critical path, it is rarely a matter of concern. It is a limitation, but it is also a motivation behind such a broad evaluation in order to understand its impact. FNs are very rare on their own and even less frequently leading to additionally missed anomalies. In the evaluation, we analyze this effect in detail and show that missed anomalies due to early stopping are extremely rare across the considered, practically motivated parameter intervals. Even if a subsequent anomaly after a FN is initially missed, it remains missed only if it has no other causal connections that are investigated via input error code (diagnostic association) or anomalous connection. This is also why it happens almost never across all experiments.

The number of FPs depends on how far γ is away from 1.0 and $n_c^{i_\eta}$ based on α and β, since they determine the number of applications of γ. The number of FNs depends on α and γ. Anomalies, which are determined by α, are a precondition for FNs, but not for FPs. The runtime is not only determined by the number and length of ground truth fault paths, but also by the length and number of determined (incorrect) fault paths (cf. perfect correlation $\rho(r_i^{max}, f_i^{max} + \tilde{f}_i^{max})$). The rationale is that all potential paths must be generated and considered, regardless of their correctness. The system's efficiency in a real-world, step-by-step diagnostic scenario is primarily contingent upon the reaction times of humans that provide the input signals, i.e., the human interactions. In cases where all data is entered into the system instantaneously, this analysis holds true. Otherwise, the runtime is determined by human interaction times, which are of course subjective and situation-based. We demonstrated that the system functions flawlessly at $\gamma = 1.0$ ($\gamma = 1.0 \implies p_2^i = 100$), i. e., perfect results of the system when everything is purely deterministic. In addition, we gave theoretical reflections on the nature of the problem and provided some guidance on where the system is still expected to meet practitioners' expectations. When $\gamma < 1.0$ (evaluation under certain circumstances), the configuration, i. e., the type of problem instance, is very decisive for the reliability of the approach. The results enable an estimation of the expected domain viability based on the parameter configuration and also some requirements based on performance expectations. In category $p_2^i \in [75, 100)\%$ with $\alpha = 0.01$, $\beta = 0.05$ and $\gamma \in [0.95, 0.99]$, the system achieves $\bar{p_0}^i = 1.0$, $\bar{p_3}^i = 0.97$ and $\bar{p_1}^i = 0.94$. This practically realistic configuration leads to almost perfect results. The next group $p_2^i \in [50, 75)\%$ contained three instance sets with $\alpha = 0.01$, $\beta \in [0.05, 0.2]$, $\gamma^{LB} \in [0.9, 0.95]$, and $\gamma^{UB} \in [0.95, 0.99]$, still leading to $\bar{p_0}^i = 1.0$, but $\bar{p_1}^i \in [0.78, 0.85]$. Moreover, $\bar{p_3}^i \in [0.92, 1.0]$. These configurations are still solved rather well. $p_2^i \in [5, 50)\%$ is based on quite a number of configurations: $\alpha \in [0.01, 0.1]$, $\beta \in [0.05, 0.2]$, $\gamma^{LB} \in [0.9, 0.95]$, and $\gamma^{UB} \in [0.95, 0.99]$. The instances are still solved with $\bar{p_0}^i \in [0.93, 1.0]$ and $\bar{p_3}^i \in [0.93, 0.98]$, but $\bar{p_1}^i = 0.47$ in the worst case, which is clearly not satisfactory from a practical point of view. After all, $p_2^i \in [0, 5)\%$ is far from being practically viable. The most extreme case goes down to $\bar{p_1}^i = 0.19$ for $i =< 10\_20\_90\_95 >$. For such a large number of anomalies and such high connectivity, such a model accuracy is inadequate. We confirmed that $\bar{n_c}^i$ was sufficient for convergence of the various metrics in each instance set. Crucially, achieving a higher recall than precision is not the result of a prioritizing mechanism built into the system, but is purely based on the considered, practically motivated parameter space. The system is intended for use in domains where anomalies are the exception. The system is generally better at predicting positives because there is simply less potential for error. Generally, it is rather unlikely to classify something incorrectly, since $\gamma_i \geq 0.9 \, \forall i \in I$, but if it happens, it is a lot less likely to predict wrong and miss an anomaly with it. We have seen that the human still plays a crucial role, e. g., in repeating (verifying) the process in case of doubt, as it is non-deterministic due to changes in the recordings. The system usually finds slightly more anomalies than expected due to FPs, sometimes it finds the exact number of expected anomalies, but never less than expected. Regarding the number of missed anomalies, we found that $\bar{m_1}^i \in [0.0, 1.9]$ and $\bar{m_2}^i \in [0.0, 0.3]$, so at most 2.2 missed anomalies in total on average. Thus, there are very few cases of entirely missed components, but there are some. Importantly, it is not only possible to miss anomalies due to FNs, but also due to TNs in combination with the abortion criterion and non-propagated errors (the latter is only true in real-world scenarios, though). We saw that a single misclassification can in principle lead to an arbitrary number of missed anomalous links, although it is unlikely to be very large depending on the domain under consideration and the degree of isolation of the components. However, if the domain in question has very long chains of components instead of a relatively dense graph as considered in this work, it might be worth keeping in mind. A further insight is that low $F1$ scores are always negative for the overall ground truth match performance, but even a perfect $\bar{F}1$ does not guarantee a high p_2^i due to missed anomalies. In other words: Good classification results are a necessary but not a sufficient condition for good end results (except for $\gamma = 1.0$). Few misclassifications can have a huge impact in terms of ground truth matches. A high $\bar{p_1}^i$ is insufficient to obtain a high p_2^i. Moreover, $p_0^{i_\eta}$ and $p_3^{i_\eta}$ are not affected by FPs and thus purely determined by FNs and early stoppings, whereas the fault paths are. If $p_0^{i_\eta}$ is better than the $F1$ score (the average $\bar{F}1$ across all instance sets is 0.9, the average $\bar{p_0}^i$ is 0.96, and the average $\bar{p_3}^i = 0.97$), it can be due to FPs. FPs negatively affect $F1$, but have no negative effect on the $p_0^{i_\eta}$ score, they can even lead to an accidental discovery of a hidden anomaly. Intuitively, the worse the model, the better the graph should be connected to compensate misclassifications. However, this also has the negative side effect of leading to more classifications and therefore more potential misclassifications and a greater impact of misclassifications. When there are more classifications or a weaker model accuracy, there is a lower $p_0^{i_\eta}$, except for $\gamma = 1.0$, then $n_c^{i_\eta}$ is irrelevant. In general, the performance is worse when there are too many classifications with a too poor model accuracy, i.e., $\frac{\bar{n_c}^i}{\gamma}$ should be small for good performance. As β increases, the number of fault paths affected by a misclassification increases. Overall, there is a tendency towards better performance with smaller α and β values. With greater connectivity due to β, there are more early stoppings or FPs, e. g., due to more classifications. A smaller $\frac{\alpha}{\beta}$ ratio seems beneficial for $p_0^{i_\eta}$, which demonstrates the compensatory effect of β. Increasing β has negative side effects for the overall performance, but these do not affect $p_0^{i_\eta}$. $p_0^{i_\eta}$ benefits from the increased $c_r^{i_\eta}$. In relation to α, larger β values improve the $p_0^{i_\eta}$ performance. The fault path length depends on the combination of α and β (cf. almost perfect correlation $\rho(\alpha\beta, l_i^a)$). More fault paths and longer fault paths, which usually coincide (cf. $\rho(f_i^a, l_i^a) = 0.81$), are harder to match completely correctly, which is why p_2^i is way worse for larger numbers of fault paths that are longer, except when having perfectly accurate models. The effect on f_i^a is way stronger for α, which is a precondition for any fault paths. β, on the other hand, is not able to affect f_i^a by itself, only in combination with α. The range of fault path deviations within one instance set is rather large, $[2, 30246]$ in the most extreme case. Each instance set has at least one instance with 0 fault path deviations, except $< 10\_20\_90\_95 >$ with at least 2. Usually, $\tilde{f}_i^{max} < 105$, in many cases even $\tilde{f}_i^{max} < 10$, but there are some extreme exceptions. $\tilde{f}_i^a$ is highest in instance sets with the longest and most fault paths with imperfect models. When dealing with imperfect γ values, it is advised to have few anomalies that are sparsely connected, or the instance will explode in terms of deviations, e. g., $i =< 10\_20\_90\_95 >$. The even more extreme case of $\alpha = \beta = 0.2$ turned out to be infeasible due to excessively large f_i^a. A high degree of connectivity can be harmful: The beneficial effect, which is that missed anomalies can be reached again via another path, seems to be canceled out by the increased $c_r^{i_\eta}$. However, the beneficial effect would be even stronger if there were less isolated anomalies, i. e., more chains of co-occurring anomalies. Nevertheless, we took a close look at how well the system is able to compensate model inaccuracies through structural knowledge. In cases where anomalies are missed due to FNs, there always seems to be enough connectivity for compensation. On the other hand, when there is not enough connectivity for compensation, it is also very unlikely that anomalies will be missed, so that no compensation is required. There is a natural balance between the two: The degree of compensation naturally scales with the number of missed anomalies. Nevertheless, higher connectivity results in higher problem complexity and usually leads to worse overall performance if the model accuracy is not enhanced accordingly. Missed chances are more likely when β is higher because then it is more likely that there is another connection, but at the same time they are less likely because then they are less likely to be missed due to the overall increased connectivity. Situations without a second chance are extremely rare across all considered instance sets based on the same reasoning. Although there are few, missed chances have a huge impact on the overall performance. There is a perfect positive correlation $\rho(\beta, \bar{c_1}^i)$ between connectivity and compensation (if α is sufficiently large, e. g., $\alpha = 0.2$). The positive compensatory effect of β is clearly visible, and it grows with the otherwise missed anomalies themselves, naturally balancing them. The growth of compensation is way stronger than the growth of misclassifications. However, it is not only the misclassifications themselves that are harmful, but also the number of fault paths that are affected due to the increased connectivity of the network. The compensation grows faster / stronger than the negative effects, but the negative effects are still significant, particularly with regard to $\bar{p_1}^i$. Consequently, the problem is not so much that there are more misclassifications, but that a misclassification has a stronger impact on the overall performance. In summary, the higher compensatory capability is accompanied by an increased potential for further misclassifications and a higher impact of single misclassifications. In a way, there is always as much compensation as required. Ultimately, it is not as simple as increasing β to compensate inaccurate models. On the contrary, the importance of high model accuracies is even more significant when dealing with highly connected diagnostic domains. A sparse network is not necessarily worse than a highly connected one when having suboptimal model accuracies. This is particularly highlighted by the very strong positive correlation $\rho(\bar{FP} + \bar{FN}, \beta)$ and the strong negative correlation $\rho(\beta, \bar{p_1}^i)$ (when $\alpha = 0.2, \gamma \in [0.95, 0.99]$). A further question was how many classifications $n_c^{i_\eta}$ can reasonably be carried out at most based on a certain γ. As always, it is advisable to optimize the models used for sensor signal evaluation to achieve the highest possible accuracy. Also, with increasing $n_c^{i_\eta}$ at least one misclassification very quickly gets close to certainty with imperfect γ. Nevertheless, we determined some approximate values for how many chained classifications (based on the connectivity of the domain) are still feasible at certain model accuracies. If $p_2^i \geq 50\%$ is expected, it is advised to have $\bar{n_c}^i < 25$, except for perfectly accurate models. If very good results are expected, i. e., $\bar{p_1}^i \geq 0.9$, it is advised to perform as few classifications as possible or to obtain close to perfectly accurate models. Obviously, it is not possible to arbitrarily reduce the number of classifications, as they are determined based on the problem definition, so it is meant as a guideline for assessing feasibility. $p_0^{i_\eta}$ is in a sense more critical as it directly measures the identified problematic links. $p_1^{i_\eta}$ can, for instance, be low due to a number of FPs, which is not that critical if all ground truth TPs are actually found. Even with the worst $\gamma = 0.92$, it is possible to go up to $n_c^{i_\eta} > 125$, i. e., $c_r^{i_\eta} = 1$, if a $\bar{p_0}^i = 0.8$ suffices. If $\bar{p_0}^i \geq 0.9$ is expected, $\gamma \geq 0.97$ is sufficient to go up to $c_r^{i_\eta} = 1$ with $|C| = 129$. However, it is still sensible to perform only as many classifications as absolutely necessary, at best $n_c^{i_\eta} < 25$, if $\gamma \in [0.93, 0.97]$. For p_2^i, which is even stricter than $p_1^{i_\eta}$, $\gamma = 0.97$ is the only imperfect model that is performing well enough to fulfill the expectation of $p_2^i \geq 70\%$, and only with an average $\bar{n_c}^i = 10.32$. If we ask for $p_2^i \geq 90\%$, only the perfectly accurate models are able to achieve this. This is a further confirmation that it is advantageous to minimize $n_c^{i_\eta}$. Overall, we saw that $\bar{p_0}^i \geq 0.85 \, \forall i \in I$, even in case of challenging configurations, which means that the considered model accuracies of $\gamma \in [0.9, 1]$ are fairly satisfying. Regarding the entire diagnosis, most instance sets $i \in I$ result in $d_s^i = 100\%$. The deviating ones all still have $d_s^i \geq 95\%$. It is not surprising that all of these share $\alpha = 0.01$. In the $\alpha = 0.01$ cases, there is only one anomaly for $|C| = 129$, which means that a single misclassification can lead to *no_diag*.

VIII. CONCLUSION AND DISCUSSION

This paper generalizes and extends our approach presented in [20] and provides a framework for multimodal, human-in-the-loop anomaly detection and complex fault diagnosis. It introduces a systematic, practically motivated, large-scale, domain-agnostic synthetic problem instance generation for diagnosis of systems with causally interconnected components that enable some form of sensory assessment per component. We formalize the abstract problem and define a parameter space covering the relevant aspects that can be configured during instance generation. With this, the properties of structures of connected systems can be altered to reflect a wide range of practical diagnosis domains. Additionally, we provide a thorough evaluation of the generalized and significantly extended version of our previously proposed hybrid neuro-symbolic diagnosis system [20]. The neuro-symbolic diagnosis benchmark goes beyond the specific use case in [20] by formalizing and generalizing the core properties that it shares with many other diagnostic domains. We explore the performance of the system across a broad spectrum of configurations and thus theoretical domains. We also analyze the impact of certain configuration aspects on the solving procedure and results. Essentially, we use ANNs to detect anomalies at components suggested by a KG based on the provided fault context. Symbolic approaches are used to guide the ANN-based classifications and to isolate the problem. The reasoning of the system is encoded in the presented state machine architecture. Apart from the general advantage of neuro-symbolic systems that both paradigms are mutually beneficial and compensate for each other's weaknesses from the introduction, we can now also say more specifically that the presented system requires the neuro-symbolic architecture and how hybridization contributes to the given task. This is because the general diagnostic problem under consideration can be solved more effectively and efficiently by a neuro-symbolic system compared to a purely connectionist or purely symbolic model.

It is unrealistic to manually formulate symbolic rules for each new sensor signal dataset, clearly justifying the introduction of neural networks. Moreover, in a purely connectionist system, there would be no entry point and an exhaustive enumeration would have to be performed leaving only the anomalies and no knowledge of fault paths and root causes. We not only answer specific questions with regard to the performance level of our diagnosis framework, but also consider fundamental aspects of the general type of diagnosis problem, e. g., the balance between compensation on the one side and increased fault potential on the other, based on the connectivity of the network. There are several tradeoffs and balancing factors that are worth considering in every domain and this paper can help guiding it. In [20], we argue that the system has the potential to drastically improve the efficiency and precision of the diagnostic process for modern vehicles, thereby helping to compensate for the shortage of specialists and preserving scarce expert knowledge through the resulting KG. In this work, we perform a systematic evaluation based on synthetic problem instances, i. e., a formalized problem structure, across all kinds of abstract diagnostic domains to assess the correct functionality and limitations of the system. We not only generalize the architecture, but also show how the system performs under different conditions. To do this, we define the core parameters to generate and solve 4000 randomized problem instances. This systematic process ensures that the developed architecture is robust and reliable not only in a specific problem case, but also across a wide range of situations and domains. This is crucial in order to transfer the system to different domains and to enable potential users to assess its practical effectiveness and limitations for their respective domains. To this end, we also explore the scalability in terms of performance and computational costs to cases with significantly larger or smaller numbers of components. Moreover, we heavily improve the transferability to other domains such as anomaly detection in industrial facilities or medical diagnosis by generalizing both the state machine and the ontology underlying the KG based on the neuro-symbolic architecture shown in Fig. 1. The presented framework provides a blueprint for addressing the problem of guiding a diagnostic process relying on both domain expertise and sensor signal interpretation. The explanatory report, contextualizing the generated heatmaps, enables domain experts to readily assess whether these areas are plausible bases for decision making. These explainability techniques not only reduce error-proneness by enabling humans to verify the result, but can also increase trust in the models – also supported by a human-in-the-loop approach. Finally, it should be mentioned that all the developed software, i.e., every module or component that contributes to the diagnosis system or its evaluation, is released as open source software (URL references are provided as footnotes in the respective sections) to facilitate reproducibility and domain transfer. To summarize, we consider the integration of machine learning and XAI into knowledge-supported fault diagnosis systems.

A concrete next step would be to consider instances or domains with anomalies that are more clustered and not so evenly distributed across the entire component space. Clusters are arguably practically plausible in many domains, as anomalies may occur in common subsystems. In this work, point anomalies were mainly considered due to the uniform distribution, as long chains are unlikely, but burst anomalies could be another interesting aspect to investigate. For this, there could be a set of instances with a few manually constructed chains of anomalies, e. g., to observe the effects of an increase in β in such cases. As anticipated, we plan to further demonstrate and analyze the architecture's potential for knowledge discovery by using the UCR time series datasets in a way where each individual dataset represents one component of an entity of diagnosis. This would not have contributed to the evaluation that was the focus of this paper, but is nevertheless a meaningful next step. Also beyond the scope of this work but a meaningful extension would be a process for automated learning from diagnostic errors. While mechanisms for updating, refining, and removing entries within the KG have been implemented, there is no automated approach to learning from diagnostic errors identified by human experts beyond enabling the basis for it via logging and contextualizing explanatory reports. However, incorrect outcomes are typically not attributable to erroneous KG entries. Typically, they refer to misclassifications that can be addressed via repetition (in case of doubt) or training more accurate classification models. If there is a problem with the KG, it is commonly due to incompleteness (missing true facts) rather than incorrectness (holding explicitly false facts). This incompleteness is a natural state under the open-world assumption and requires extension via the associated pool of human experts. Obviously, this is assuming a KG of high correctness due to the construction based on expert knowledge as in [20]. An arbitrary incorrect KG can result in arbitrarily impaired solutions. In the end, unsuccessful diagnoses are logged in the KG and can be analyzed by human experts. In future work, we also plan to instantiate the system in a robotic self-diagnosis application to further gather real-world data and develop mechanisms for automated conclusions based on unsuccessful diagnoses.

ACKNOWLEDGMENT

The authors express their gratitude to the anonymous reviewers for their constructive feedback and suggestions, which helped to improve the quality of this article.

REFERENCES

[1] R. Milne and C. Nicol, "TIGER: Continuous diagnosis of gas turbines," in *Proc. ECAI*, 2000, pp. 711–715.

[2] F. Puppe, M. Atzmueller, G. Buscher, M. Huettig, H. Lührs, and H.-P. Buscher, "Application and evaluation of a medical knowledge-system in sonography (SonoConsult)," in *Proc. ECAI*, 2008, pp. 683–687.

[3] S. P. Kavulya, K. Joshi, F. D. Giandomenico, and P. Narasimhan, "Failure diagnosis of complex systems," in *Resilience Assessment and Evaluation of Computing Systems*. Springer, 2012, pp. 239–261. [Online]. Available: https://dblp.org/db/books/collections/wolter2012.html#KavulyaJGN12

[4] F. Hayes-Roth, D. A. Waterman, and D. B. Lenat, *Building Expert Systems*. Reading, MA, USA: Addison-Wesley, 1983.

[5] F. Puppe, *Systematic Introduction to Expert Systems*. Cham, Switzerland: Springer, 1993.

[6] M. Stefik, *Introduction to Knowledge Systems*. San Mateo, CA, USA: Morgan Kaufmann, 1995.

[7] C. Nan, F. Khan, and M. T. Iqbal, "Real-time fault diagnosis using knowledge-based expert system," *Process Saf. Environ. Protection*, vol. 86, no. 1, pp. 55–71, Jan. 2008.

[8] W. Li, H. Li, S. Gu, and T. Chen, "Process fault diagnosis with model- and knowledge-based approaches: Advances and opportunities," *Control Eng. Pract.*, vol. 105, Dec. 2020, Art. no. 104637.

[9] Y. Chi, Y. Dong, Z. J. Wang, F. R. Yu, and V. C. M. Leung, "Knowledge-based fault diagnosis in industrial Internet of Things: A survey," *IEEE Internet Things J.*, vol. 9, no. 15, pp. 12886–12900, Aug. 2022.

[10] B. Steenwinckel, P. Heyvaert, D. De Paepe, O. Janssens, S. V. Hautte, A. Dimou, F. De Turck, S. Van Hoecke, and F. Ongenae, "Towards adaptive anomaly detection and root cause analysis by automated extraction of knowledge from risk analyses," in *Proc. SSN Workshop*, vol. 2213, 2018, pp. 17–31.

[11] I. Kononenko, "Machine learning for medical diagnosis: History, state of the art and perspective," *Artif. Intell. Med.*, vol. 23, no. 1, pp. 89–109, Aug. 2001.

[12] M. Klemettinen, H. Mannila, and H. Toivonen, "A data mining methodology and its application to semi-automatic knowledge acquisition," in *Proc. DEXA*, Sep. 1997, pp. 670–677.

[13] H. Leitich, K.-P. Adlassnig, and G. Kolarz, "Evaluation of two different models of semi-automatic knowledge acquisition for the medical consultant system CADIAG-II/RHEUMA," *Artif. Intell. Med.*, vol. 25, no. 3, pp. 215–225, Jul. 2002.

[14] W. Shi, J. A. Barnden, M. Atzmueller, and J. Baumeister, *An Intelligent Diagnosis System Handling Multiple Disorders*. Cham, Switzerland: Springer, 2005, pp. 421–430.

[15] M. Atzmueller, J. Baumeister, and F. Puppe, "Semi-automatic learning of simple diagnostic scores utilizing complexity measures," *Artif. Intell. Med.*, vol. 37, no. 1, pp. 19–30, May 2006.

[16] W. Yun, X. Zhang, Z. Li, H. Liu, and M. Han, "Knowledge modeling: A survey of processes and techniques," *Int. J. Intell. Syst.*, vol. 36, no. 4, pp. 1686–1720, Apr. 2021.

[17] D. Gunning, M. Stefik, J. Choi, T. Miller, S. Stumpf, and G.-Z. Yang, "XAI—Explainable artificial intelligence," *Sci. Robot.*, vol. 4, Jan. 2019, Art. no. eaay7120.

[18] O. Biran and C. Cotton, "Explanation and justification in machine learning: A survey," in *Proc. IJCAI Workshop XAI*, 2017, pp. 8–13.

[19] A. Barredo Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. Garcia, S. Gil-Lopez, D. Molina, R. Benjamins, R. Chatila, and F. Herrera, "Explainable artificial intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI," *Inf. Fusion*, vol. 58, pp. 82–115, Jun. 2020.

[20] T. Bohne, A. K. P. Windler, and M. Atzmueller, "A neuro-symbolic approach for anomaly detection and complex fault diagnosis exemplified in the automotive domain," in *Proc. K-CAP*, Dec. 2023, pp. 35–43.

[21] F. Puppe, "Knowledge reuse among diagnostic problem-solving methods in the shell-kit D3," *Int. J. Hum.-Comput. Stud.*, vol. 49, no. 4, pp. 627–649, Oct. 1998.

[22] R. Studer, D. Fensel, S. Decker, and V. R. Benjamins, "Knowledge engineering: Survey and future directions," in *Proc. 5th Biannual German Conf. Knowl.-Based Syst.*, 1999, pp. 1–23.

[23] J. Baumeister and A. Striffler, "Knowledge-driven systems for episodic decision support," *Knowledge-Based Syst.*, vol. 88, pp. 45–56, Nov. 2015.

[24] Z. Gao, C. Cecati, and S. X. Ding, "A survey of fault diagnosis and fault-tolerant techniques—Part I: Fault diagnosis with model-based and signal-based approaches," *IEEE Trans. Ind. Electron.*, vol. 62, no. 6, pp. 3757–3767, Jun. 2015.

[25] M. Atzmueller, B. Klöpper, H. A. Mawla, B. Jäschke, M. Hollender, M. Graube, D. Arnu, A. Schmidt, S. Heinze, L. Schorer, A. Kroll, and G. Stumme, "Big data analytics for proactive industrial decision support," *atp Ed.*, vol. 58, no. 9, p. 62, Sep. 2016.

[26] D. Tole and N. Joshi, "Simplifying data preparation for analysis using an ontology for machine data," in *Proc. 10th Int. Joint Conf. Knowl. Discovery, Knowl. Eng. Knowl. Manage.*, 2018, pp. 167–174.

[27] A. Behravan, S. Meckel, and R. Obermaisser, "Generic fault-diagnosis strategy based on diagnostic directed acyclic graphs using domain ontology in automotive applications," in *Proc. Symp. AmE*, Mar. 2019, pp. 1–5.

[28] V. D. Majstorović, "Expert systems for diagnosis and maintenance: The state-of-the-art," *Comput. Ind.*, vol. 15, nos. 1–2, pp. 43–68, Jan. 1990.

[29] R. Reinertsen, "Residual life of technical systems; diagnosis, prediction and life extension," *Rel. Eng. Syst. Saf.*, vol. 54, no. 1, pp. 23–34, Oct. 1996.

[30] Y. Lei, B. Yang, X. Jiang, F. Jia, N. Li, and A. K. Nandi, "Applications of machine learning to machine fault diagnosis: A review and roadmap," *Mech. Syst. Signal Process.*, vol. 138, Apr. 2020, Art. no. 106587.

[31] Z. S. Chen, Y. M. Yang, and Z. Hu, "A technical framework and roadmap of embedded diagnostics and prognostics for complex mechanical systems in prognostics and health management systems," *IEEE Trans. Rel.*, vol. 61, no. 2, pp. 314–322, Jun. 2012.

[32] A. Ishiguro, Y. Watanabe, and Y. Uchikawa, "Fault diagnosis of plant systems using immune networks," in *Proc. IEEE Int. Conf. MFI 94. Multisensor Fusion Integr. Intell. Syst.*, Oct. 1994, pp. 34–42.

[33] V. Alcaraz-González, R. H. López-Bañuelos, J.-P. Steyer, H. O. Méndez-Acosta, V. González-Álvarez, and C. Pelayo-Ortiz, "Interval-based diagnosis of biological systems—A powerful tool for highly uncertain anaerobic digestion processes," *CLEAN-Soil, Air, Water*, vol. 40, no. 9, pp. 941–949, Sep. 2012.

[34] R. Fezai, K. Abodayeh, M. Mansouri, H. Nounou, and M. Nounou, "Fault diagnosis of biological systems using improved machine learning technique," *Int. J. Mach. Learn. Cybern.*, vol. 12, no. 2, pp. 515–528, Feb. 2021.

[35] H.-P. Eich and C. Ohmann, "Internet-based decision-support server for acute abdominal pain," *Artif. Intell. Med.*, vol. 20, no. 1, pp. 23–36, Sep. 2000.

[36] H.-P. Buscher, C. Engler, A. Führer, S. Kirschke, and F. Puppe, "HepatoConsult: A knowledge-based second opinion and documentation system," *Artif. Intell. Med.*, vol. 24, no. 3, pp. 205–216, Mar. 2002.

[37] H. Ismail Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P.-A. Müller, "Deep learning for time series classification: A review," *Data mining Knowl. discovery*, vol. 33, no. 4, pp. 917–963, 2019.

[38] Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou, "A survey of convolutional neural networks: Analysis, applications, and prospects," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 12, pp. 6999–7019, Dec. 2022.

[39] D. Hawkins, *Identification of Outliers*. London, U.K.: Chapman & Hall, 1980.

[40] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly Detection: A Survey," *ACM Comput. Surv.*, vol. 41, pp. 1–58, Jan. 2009.

[41] M. Atzmueller, D. Arnu, and A. Schmidt, "Anomaly detection and structural analysis in industrial production environments," in *Proc. IDSC*, 2017, pp. 91–95.

[42] R. Chalapathy and S. Chawla, "Deep learning for anomaly detection: A survey," 2019, *arXiv:1901.03407*.

[43] G. Pang, C. Shen, L. Cao, and A. van den Hengel, "Deep learning for anomaly detection: A review," *ACM Comput. Surveys*, vol. 54, pp. 1–38, Jan. 2021.

[44] Y. Luo, Y. Xiao, L. Cheng, G. Peng, and D. Yao, "Deep learning-based anomaly detection in cyber-physical systems: Progress and opportunities," *ACM Comput. Surv.*, vol. 54, no. 5, pp. 1–36, Jun. 2022.

[45] Y. Wang, Q. Han, G. Zhao, M. Li, D. Zhan, and Q. Li, "A deep neural network based method for magnetic anomaly detection," *IET Sci., Meas. Technol.*, vol. 16, no. 1, pp. 50–58, Jan. 2022.

[46] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 618–626.

[47] M. He, B. Li, and S. Sun, "A survey of class activation mapping for the interpretability of convolution neural networks," in *Proc. ICSINC*, 2023, pp. 399–407.

[48] A. Theissler, "Detecting anomalies in multivariate time series from automotive systems," Ph.D. dissertation, School of Eng. and Design, Brunel Univ., Uxbridge, U.K., 2013.

[49] T. R. Besold, A. S. d'Avila Garcez, S. Bader, H. Bowman, P. M. Domingos, P. Hitzler, K. Kühnberger, L. C. Lamb, P. M. V. Lima, L. de Penning, G. Pinkas, H. Poon, and G. Zaverucha, "Neural-symbolic learning and reasoning: A survey and interpretation," in *Neuro-Symbolic Artificial Intelligence: The state of the Art* (Frontiers in Artificial Intelligence and Applications), vol. 342. Amsterdam, The Netherlands: IOS Press, 2021, pp. 1–51.

[50] P. Hitzler, A. Eberhart, M. Ebrahimi, M. K. Sarker, and L. Zhou, "Neuro-symbolic approaches in artificial intelligence," *Nat. Sci. Rev.*, vol. 9, no. 6, p. 035, Jun. 2022.

[51] H. Liu, R. Ma, D. Li, L. Yan, and Z. Ma, "Machinery fault diagnosis based on deep learning for time series analysis and knowledge graphs," *J. Signal Process. Syst.*, vol. 93, no. 12, pp. 1433–1455, Dec. 2021.

[52] A. Hogan, E. Blomqvist, M. Cochez, C. d'Amato, G. de. Melo, C. Gutiérrez, S. Kirrane, J. E. L. Gayo, R. Navigli, S. Neumaier, A.-C. N. Ngomo, A. Polleres, S. M. Rashid, A. Rula, L. Schmelzeisen, J. Sequeda, S. Staab, and A. Zimmermann, "Knowledge graphs," *ACM Comput. Surv.*, vol. 54, no. 4, pp. 1–37, 2021.

[53] S. Ji, S. Pan, E. Cambria, P. Marttinen, and P. S. Yu, "A survey on knowledge graphs: Representation, acquisition, and applications," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 2, pp. 494–514, Feb. 2022.

[54] T. Aravanis and I. Kabouris, "A neuro-symbolic approach for fault diagnosis in smart power grids," in *Proc. 26th Pan-Hellenic Conf. Informat.*, Nov. 2022, pp. 90–95.

[55] G. Brewka, T. Eiter, and M. Truszczyński, "Answer set programming at a glance," *Commun. ACM*, vol. 54, no. 12, pp. 92–103, Dec. 2011.

[56] Q. Li, Y. Liu, S. Sun, Z. Qin, and F. Chu, "Deep expert network: A unified method toward knowledge-informed fault diagnosis via fully interpretable neuro-symbolic AI," *J. Manuf. Syst.*, vol. 77, pp. 652–661, Dec. 2024.

[57] H. Kautz, "The third AI summer: AAAI Robert S. Engelmore memorial lecture," *AI Mag.*, vol. 43, no. 1, pp. 93–104, Mar. 2022.

[58] Z. Wang, W. Yan, and T. Oates, "Time series classification from scratch with deep neural networks: A strong baseline," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, May 2017, pp. 1578–1585.

[59] K. Fauvel, T. Lin, V. Masson, É. Fromont, and A. Termier, "XCM: An explainable convolutional neural network for multivariate time series classification," *Mathematics*, vol. 9, no. 23, p. 3137, Dec. 2021.

[60] A. Holzinger, "Human–computer interaction and knowledge discovery (HCI-KDD): What is the benefit of bringing those two fields to work together?" in *Proc. CD-ARES*, 2013, pp. 319–328.

[61] M. Atzmueller, "Declarative aspects in explicative data mining for computational sensemaking," in *Proc. DECLARE*, 2018, pp. 97–114.

[62] R. L. Draelos and L. Carin, "Use HiResCAM instead of grad-CAM for faithful explanations of convolutional neural networks," 2020, *arXiv:2011.08891*.

[63] A. Chattopadhay, A. Sarkar, P. Howlader, and V. N. Balasubramanian, "Grad-CAM++: Generalized gradient-based visual explanations for deep convolutional networks," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2018, pp. 839–847.

[64] H. Wang, Z. Wang, M. Du, F. Yang, Z. Zhang, S. Ding, P. Mardziel, and X. Hu, "Score-CAM: Score-weighted visual explanations for convolutional neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2020, pp. 111–119.

[65] D. Smilkov, N. Thorat, B. Kim, F. Viégas, and M. Wattenberg, "Smooth-Grad: Removing noise by adding noise," 2017, *arXiv:1706.03825*.

[66] P.-T. Jiang, C.-B. Zhang, Q. Hou, M.-M. Cheng, and Y. Wei, "LayerCAM: Exploring hierarchical class activation maps for localization," *IEEE Trans. Image Process.*, vol. 30, pp. 5875–5888, 2021.

[67] J. Bohren and S. Cousins, "The SMACH high-level executive," *IEEE Robot. Autom. Mag.*, vol. 17, no. 4, pp. 18–20, Apr. 2011.

[68] T. Bohne, A.-K. P. Windler, and M. Atzmueller, "Saliency map-guided knowledge discovery for subclass identification with LLM-based symbolic approximations," 2025, *arXiv:2511.07126*.

[69] J. Ott, A. Ledaguenel, C. Hudelot, and M. Hartwig, "How to think about benchmarking neurosymbolic AI?" in *Proc. NeSy*, 2023, pp. 248–254.

[70] F. Yang and D. Xiao, "Progress in root cause and fault propagation analysis of large-scale industrial processes," *J. Control Sci. Eng.*, vol. 2012, pp. 1–10, Jan. 2012.

[71] S. A. A. Taqvi, H. Zabiri, F. Uddin, M. Naqvi, L. D. Tufa, M. Kazmi, S. Rubab, S. R. Naqvi, and A. S. Maulud, "Simultaneous fault diagnosis based on multiple kernel support vector machine in nonlinear dynamic distillation column," *Energy Sci. Eng.*, vol. 10, no. 3, pp. 814–839, Mar. 2022.

TIM BOHNE received the B.Sc. and M.Sc. degrees in computer science from Osnabrück University, Germany, in 2019 and 2022, respectively, where he is currently pursuing the Ph.D. degree with the Semantic Information Systems Group. From 2017 to 2022, he worked as a Research Assistant with the Department of Plan-Based Robot Control, German Research Center for Artificial Intelligence. Concurrently, from 2019 to 2020, he was employed as a Research Assistant with the Combinatorial Optimization Group, Osnabrück University. Since 2022, he has been a Researcher with Department of Cooperative and Autonomous Systems, German Research Center for Artificial Intelligence. He leads research projects focused on automated fault diagnosis and decision support in industrial domains with emphasis on sustainability. His research interests center around neuro-symbolic computing, with a particular focus on neuro-symbolic methods for anomaly detection and fault diagnosis.

ANNE-KATHRIN PATRICIA WINDLER received the B.Sc. and M.Sc. degrees in cognitive science from Osnabrück University, in 2018 and 2022, respectively. She has been a Researcher with the Cooperative and Autonomous Systems Research Department, German Research Center for Artificial Intelligence (DFKI), Osnabrück, since 2022. Her research interests include deep learning, time series classification, data augmentation, neuro-symbolic diagnosis, and interpretable AI.

MARTIN ATZMUELLER received the M.Sc. (Diplom-Informatiker Univ.) and Ph.D. (Dr. rer. nat.) degrees in computer science from the University of Würzburg, Germany, in 2002 an 2006, respectively, and the Habilitation (Dr.habil.) degree in computer science from the University of Kassel, Germany, in 2013.

From 2010 to 2017, he was a Senior Researcher at the University of Kassel. Subsequently, he held positions as an Associate Professor with Tilburg University, The Netherlands, and as a Visiting Professor with Université Sorbonne Paris Nord, France. In 2021, he was appointed as a Full Professor with Osnabrück University, Germany. He is a Full Professor with the Institute of Computer Science, Osnabrück University, Germany, where he heads the Semantic Information Systems Group. He is the Scientific Director with the Research Department Cooperative and Autonomous Systems, German Research Center for Artificial Intelligence (DFKI), Osnabrück, the Founding Spokesperson of the Joint Lab on Artificial Intelligence and Data Science, and a member of the Research Center Data Science, Osnabrück University. His research interests include artificial intelligence (AI), data science, and integrative AI systems, where his particular research interests include modeling complex data, explainable AI, interpretability, machine perception, and semantic modeling. This also relates to applications in trusted AI system design, in particular, relating to robot control, and integrative sensor-based AI systems.