

Semi-automatic Generation of Multiple-Choice Tests from Mentions of Semantic Relations

Renlong Ai, Sebastian Krause, Walter Kasper, Feiyu Xu, Hans Uszkoreit

German Research Center for Artificial Intelligence

Alt-Moabit 91c, 10559 Berlin, Germany

{renlong.ai, skrause, kasper, feiyu, uszkoreit}@dfki.de

Abstract

We propose a strategy for the semi-automatic generation of learning material for reading-comprehension tests, guided by semantic relations embedded in expository texts. Our approach combines methods from the areas of information extraction and paraphrasing in order to present a language teacher with a set of candidate multiple-choice questions and answers that can be used for verifying a language learners reading capabilities. We implemented a web-based prototype showing the feasibility of our approach and carried out a pilot user evaluation that resulted in encouraging feedback but also pointed out aspects of the strategy and prototype implementation which need improvements.

1 Introduction

Computer assisted language learning (CALL) opens many new opportunities for language learners and teachers. In this paper, we focus on one often used tool of this area: reading-comprehension tests. Such tests are an important means for assessing a learner's current skill level by verifying his understanding of foreign-language texts. Some work in the area of CALL has focused on reducing the teacher's workload in context of reading-comprehension tests by inventing methods for the automatic scoring of such tests (see Section 5). In contrast, we propose a strategy for the semi-automatic *generation* of learning material for reading-comprehension tests, guided by semantic relations embedded in expository text. The multiple-choice exercises ask learners to choose from a list of statements about semantic relations the one which is actually expressed in a long free text. The exercises attempt to test whether the learners understand the text and have enough language

knowledge for recognizing variants of expressions for the same semantic relations.

Our strategy combines technologies from different branches of NLP. A standard information extraction (IE) system is utilized to automatically recognize relevant entities and semantic relations among them in texts. The resulting mentions are used for the creation of (a) paraphrases of the actually mentioned facts and (b) natural-language statements expressing facts *not* mentioned in the original text, i. e., the sentence generation system takes linguistic patterns filled with entities as input and produces paraphrases as potential answer candidates. The multiple-choice exercises generated in this way are then presented to a language teacher, who has to go through them and can reject a subset of these or can replace individual elements. This human-in-the-loop step is necessary because of the noise inherent to current NLP systems. We let the teacher choose the appropriate trade-off between correctness and content-coverage of the generated (candidate) questionnaires.

The proposed strategy is implemented in a web-based prototype system, with separate interfaces for teachers (for the preparation of exercises) and learners (for conducting the exercises). This prototype is capable of handling a number of selected semantic relations from the biographic and financial domains, illustrating the applicability of the approach for the frequent class of news articles from tabloid press and business news. Based on this prototype, we carried out a pilot user study to gather insights on the best directions for future development.

In summary, the contributions of this paper are as follows:

- A strategy for the semi-automatic generation of learning material for fact-centric reading-comprehension tests.
- A way of incorporating the idea of a human-

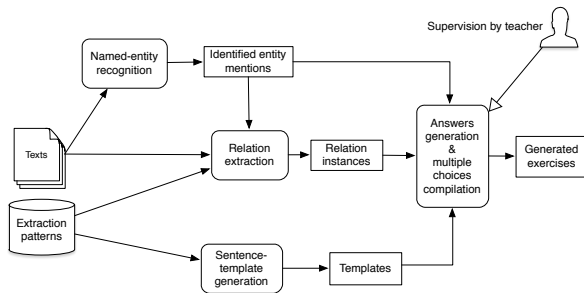


Figure 1: Workflow of exercise generation.

in-the-loop into the data flow of the strategy.

- A web-based prototype implementation of this approach, along with a pilot user study to explore directions for further development.

2 A workflow for automatic exercise generation

In this section, we present our generic approach towards the generation of candidate multiple-choice questions for given expository texts. Furthermore, we detail the steps necessary for a teacher to compile actual reading-comprehension exercises suitable for presentation to learners, given only our approach’s resulting candidates.

2.1 Reading-comprehension exercises

The reading-comprehension exercises generated by our approach ask for relational facts mentioned in a text. In order to automatically identify these, we apply a series of processing steps, as depicted in Figure 1. In the first step, the input texts, e.g., news articles, are processed by a standard component for named-entity recognition (e.g., the well-known Stanford Named Entity Recognizer by Finkel et al. (2005)), in order to identify persons, organization, locations, etc. mentioned in the text, followed by application of a relation extraction system for the identification of facts.

The information about mentioned facts and entities is passed on to a further processing step in which a mentioned instance of a semantic relation is transformed into a natural-language statement, paraphrasing the original occurrence of the fact. Furthermore, the information about named-entity occurrences is used to create false statements about relations between the entities. For each fact identified in a text, four choices are provided as potential statements about the text, only one of them stating a fact actually mentioned.

2.2 Relation Extraction

A key part of our approach is the application of a *pattern-based* relation extraction (RE) system. Such systems, e.g., NELL (Carlson et al., 2010; Mitchell et al., 2015), PATTY (Nakashole et al., 2012), DARE (Xu et al., 2007), rely on lexico-syntactic patterns that pose restrictions on the surface level or grammatical level of sentences. Their underlying assumption is that whenever a given sentence matches a given pattern (i.e., a sentence template), the sentence expresses the pattern’s corresponding semantic relation. This assumption does not always hold, hence the system output usually contains a certain amount of noise, which makes a human-in-the-loop necessary for high-precision applications.

Typically, RE systems associate patterns with a confidence score of some kind, allowing downstream components to trade precision for recall. At this step in our pipeline, we extract all the information the RE system can deliver, associate it with the extracting pattern’s score, and pass it on to the next step.

One important aspect to consider is the amount of information such an approach can extract from texts. We believe that pattern-based RE systems provide enough facts for our approach as in principle any semantic relation between entities, such as kinship relations, can be detected. For example, given the following sentence, RE systems could extract the relation instance *marriage*(Madonna, Guy Ritchie) and pass it on to the next component:

Example 1: As the skirls of a lone bagpiper gave way to the music of French pianist Katia Labèque and a local organist, the wedding ceremony of Madonna Louise Ciccone, 42, and film director Guy Ritchie, 32, began.

2.3 Answer Generation

Given the relation instances and arguments identified in the previous step, candidates for questions and answers are automatically generated by filling arguments into sentence templates. These templates are created based on patterns that were used for relation extraction in the previous step, i.e., RE patterns are utilized for two purposes in our approach.

Depending on the specific kind of RE pattern, this step involves a few straight-forward processing steps, e.g., for the case of surface-level RE patterns it involves restoring correct inflections of poten-

tially lemmatized lexical pattern elements; for the case of dependency-grammar based patterns it additionally includes a step of tree linearization, see, e.g., (Wang and Zhang, 2012).

In the following, we present some example sentence templates, used for the generation of multiple-choice tests¹:

- *marriage* relation:
 - person tied the knot with person.
 - person and person were married.
- *parent-child* relation:
 - person was raised by parent.
 - parent passed on the family gene to person.
- *foundation* relation:
 - company was founded by person.
 - person set up the first company (in location) .

A multiple-choice question is generated for every identified relation mention involving two entities, where the questions are rather generic, e.g., “Which one of the following four facts can be inferred from the text?”. The respective correct answer is generated by filling the relation instance’s arguments into a sentence template associated with the target relation. For the sentence in Example 1, a generated correct answer could be: “Madonna Louise Ciccone and Guy Ritchie were in a wonderful marriage relation”.

Wrong answers are generated on the one hand by filling the arguments in templates for other target relations. For the case of the *parent-child* relation this yields “Madonna Louise Ciccone passed on the family gene to Guy Ritchie.”. The second way wrong answers are created is by mixing in arguments from other relation instances, e.g., “Madonna Louise Ciccone tied the knot with John Ritchie”.

To avoid the generation of answer options which are easy to identify as being made up, we use only entities for wrong answers which have at least one relationship with another entity mentioned in the same article. This means that, for the example scenario outlined by Example 1, celebrities who are only mentioned once, e.g., in a list of wedding guests, are not utilized.

As another measure to improve the quality of the wrong answer options, we ensure that the respective entities are mentioned relatively close to one

another in the source text. The best case would be that they appear in the same sentence from which a relationship is extracted. Consider the following example sentence from which the instance *marriage*(Madonna, Ritchie) is extracted:

Example 2:

If Penn was Madonna’s temperamental match and boyfriend Carlos Leon, father of Lourdes, her physical ideal, Ritchie --- who reportedly calls his new wife ‘Madge’ in private --- is a man who holds his own against his high-powered bride.

Here, the approach would generate the following answer options:

- Madonna and Ritchie had a wedding. (*correct*)
- Madonna tied the knot with Carlos Leon.
- Carlos Leon passed on the family gene to Ritchie.
- Lourdes was brought up by Penn.

In order to identify the correct statement, learners need sufficient knowledge of both vocabulary and grammar, also they need to be able to resolve coreference relations between occurring entities.

Paraphrasing In order to create both challenging and motivating tests for language learners, the generated statements need to present the user with a large variety of ways to refer to semantic relations, i.e., repetitions should be avoided. We ensure this first of all by employing web-scale RE-pattern sets as a source for the sentence templates, were these sets often contain hundreds of different ways to express a given target relation (see Section 3).

To create even more variations, we also introduce paraphrasing technology to the system to reorder words in the patterns learned by relation extraction systems and produce a new sentence with the same meaning. For example, a sentence template “wife had a kid from husband” is formed from one of the patterns used in the *marriage* relation. The paraphrasing engine takes this template as input and provides templates with the same words in natural language, e.g., “from husband wife had a kid”. Both templates are treated as valid and randomly chosen to create answers.

2.4 Human supervision

As already noted earlier, employing automatic information-extraction methods has the disadvantage of inevitable noise in the system output.

¹Items with sans-serif font represent entity placeholders.

Choose the amount of the relationships you want to extract. The more relationships you get, the less accuracy they have.

Extracted relationships: 11 Generate!

local organist, the wedding ceremony of Madonna Louise Ciccone, 42, and film director Guy Ritchie, 32, began.

Madonna's 4-year-old daughter, Lourdes, shoeless and draped in a long ivory dress with short sleeves and a high neck, led the processional. Descending the staircase—its balustrade laced with ivy and white orchids—she tossed handfuls of red rose petals from a basket, almost exhausting her supply by the time she reached the front row, where she sat in her nanny's lap. "As soon as they saw Madonna's daughter throwing rose petals," says a guest, "people were crying."

Ritchie followed, turning guests' heads as he entered in a vibrant teal blazer, a Hunting Mackintosh plaid kilt of navy and green and antique diamond cufflinks, a wedding gift from his bride-to-be. (Sitting quietly nearby in the arms of a nanny, the couple's 4-month-old son, Rocco, wore a matching kilt.) The two best men—**Matthew Vaughn**, who produced Ritchie's two films, and London nightclub owner **Piers Adam**—followed close behind Ritchie, trailed by maid of honor **Stella McCartney**, who wore a gray-and-beige silk pants outfit of her own design.

As the bride ascended the stairs, says a guest, "everyone kind of gasped. She looked like a princess." Madonna, who had assumed a '50s look for her first visit to the altar with **Sean Penn** in 1985, this time cast back more than a century. Her strausless ivory silk

3. person_parent

Lourdes	person
Madonna Louise Ciccone	parent

accept decline show relations

< 1 2 3 4 5 6 7 8 9 10 11 >

Figure 2: Interface for teacher step 1.

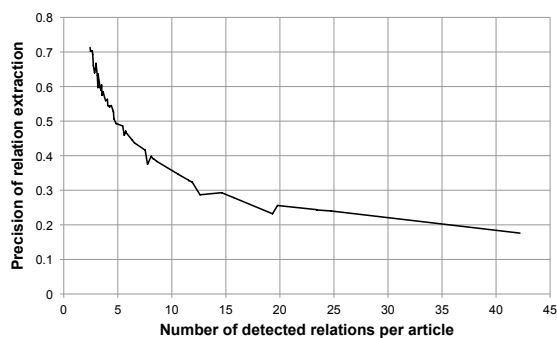


Figure 3: Precision/productivity trade-off

Given that the targeted users of the reading-comprehension exercises are language learners, it is necessary to include a step with human supervision into the data flow of our proposed system. All sentence templates, including those formulated from RE patterns and their variants created by the paraphrasing engine, are verified by teachers. Furthermore, teachers check the extracted relation instances to ensure they are actually mentioned in the text, then they verify the generated answers wrt. to grammaticality and adequacy for the context.

3 Prototype implementation

We have implemented a prototype of our system in order to test the feasibility of our proposed approach and to gather insights on future research directions, by carrying out user studies with it. The system is implemented as a browser-based application.

In principle, the approach can handle arbitrary texts. We tested it on a corpus of 140 English news

articles (Krause et al., 2014), and measured the productivity of our approach for automatic question and answer generation. For these first experiments, we used the available gold-standard entity annotation. For the relation-extraction part, we applied the RE patterns of Moro et al. (2013) to automatically extract the relations between the annotated entities in the text. These patterns are based on the dependency-grammar analysis of sentences and were extracted from a large web corpus, hence they should provide enough variation for both the detection of relation mentions in texts as well as the generation of statements about such identified mentions. We used the patterns for three kinship relations in this experiment, namely the relations *marriage*, *parent-child*, *siblings*. As a means of automatic noise reduction, we work with a combination of training-support-based pattern filters and ones relying on the distribution of relation-relevant word senses in a lexico-semantic resource, as provided by Moro et al. (2013).

The paraphrasing engine in (Ai et al., 2014) is used in our system to generate sentence variants for the patterns, part of the process involves the utilization of the sentence generator by Wang and Zhang (2012), which produces linearizations for the dependency-tree-based patterns.

To reduce a teacher's work in examining the generated exercises, we provide a two-step user interface. In the first step, extracted relation instances for a given text are displayed and require validation by the user, as shown in Figure 2. The teacher can adjust the pattern-filter parameters in order to trade precision for recall, by moving the slider in the UI. Extracted relationships are shown below the text and teachers need to go through each of them, either accept them or decline them.

By choosing different parameter values for the filters, the number of relationships found by the extractor varies. The result of this trade-off for the employed corpus is illustrated in Figure 3. If the teacher tunes the relation-extraction component to its strictest setting, approximately three relation instances per article are found, out of which two are correct. If a user is willing to invest more time into question validation (i.e., the next step), it is possible to get more than twice as much facts of a lower average accuracy, hence a teacher would need more time to examine them.

In the second step of the teacher sub-workflow, generated questions and answers are presented for

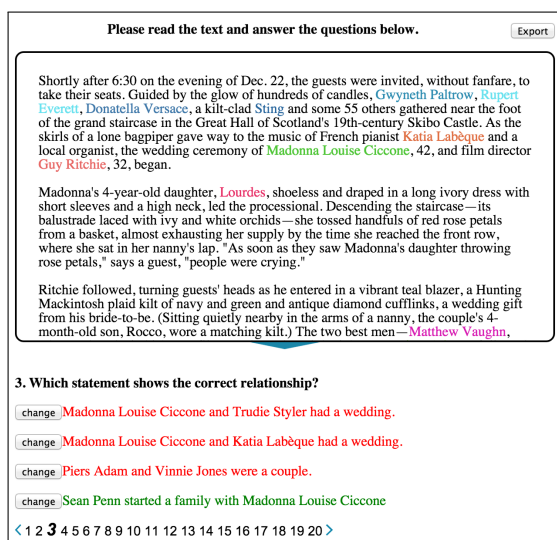


Figure 4: Interface for teacher step 2.

all previously accepted relation instances, see Figure 4. Teachers can check the answers for each question, and change them if they are, e.g., not consistent with the context induced by the article.

When teachers have verified all the questions and answers, they can press the *export* button to generate reading-comprehension exercises as shown in Figure 5. This is the interface that language learners use to interact with the system, i.e., access the teacher-approved exercises.

In order to find out the correct statements, learners need to firstly understand the semantic relations among the entities expressed in the texts and secondly have sufficient linguistic knowledge to understand the answer candidates which are paraphrases of the original sentences mentioned in the text. The paraphrases are namely linguistic variants at word (e.g., synonyms) or word-order level (e.g., topicalization). The interface provides feedback to the learners by marking the selected choice with green or red color depending on the correctness. In case a wrong answer is selected, the correct answer is shown to learners in green. If learners need more explanation, they can choose to click the *hint* button, which highlights the sentence with the relation instance mentioned in the correct answer. Furthermore, the system provides a visualization function which displays a graph with all recognized relations among the entities in the text.

4 Pilot user study

Two aspects of the implemented prototype were evaluated in separated tests with human subjects,

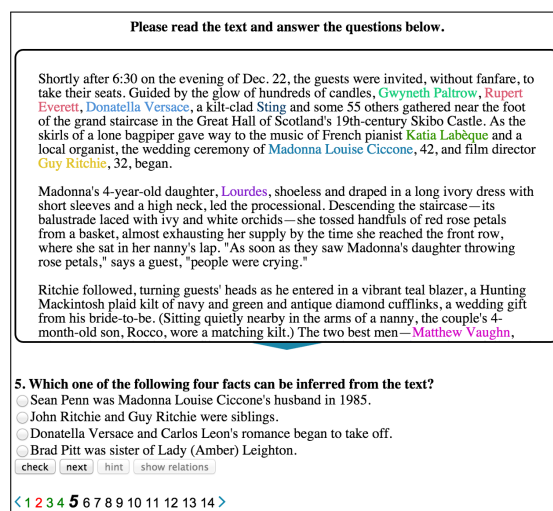


Figure 5: Example exercise, as generated by the prototype.

i.e., the interface for language learners and the interface for teachers. For the tests with learners, our interests were to find out whether:

- ... the generated multiple-choice exercises fit the learners' expectations, e.g., with respect to user friendliness.
- ... the questions are of sufficient complexity, i.e., a learner's reading comprehension skills are actually tested.
- ... the system feedback after a wrong answer does help learners in figuring out the right answer.

For the tests with teachers, our evaluation tries to determine:

- ... if the prototype provides a user-friendly interface to generate exercises from texts.
- ... how teachers think about the step-by-step generation of exercises and if the teachers' requirements are met.
- ... if teachers agree such exercises would help users achieve their language-learning goals.

We set up a field test for users in which we asked them to work with the respective interface in an online version of the prototype and to fill out a provided online questionnaire after the test. Questions in the interview for usability and acceptability are composed based on ISO NORM 9241/10,

which checks compliance to ergonomic requirements for screen work places, for example self-descriptiveness, controllability, conformity with user expectations and suitability for individualization. For this pilot study, we had five students act as teachers and language learners and asked them to take the questionnaire.

The following table lists the questions used in the interview:

The interface gives a clear concept of what there is to do.	5-Step Likert Scale (Agreement)
I can adjust the layout to suit my preferences.	5-Step Likert Scale (Agreement)
Generally, I feel no challenge in answering the questions.	5-Step Likert Scale (Agreement)
It is possible to answer the questions without fully understanding the article text, e.g. by concluding the correct answer from certain properties of the text.	5-Step Likert Scale (Agreement)
I can easily tell apart the correct answer from the wrong ones, without looking at the article text at all.	5-Step Likert Scale (Agreement)
The "hint" function makes it easier to figure out the correct answer.	5-Step Likert Scale (Agreement)
What has to be changed?/What did you like?	Open

Table 1: Questionnaire for learners.

The interface provides self-explained instructions.	5-Step Likert Scale (Agreement)
I can easily check the validity of the extracted relationships from the corresponding sentences in the article.	5-Step Likert Scale (Agreement)
I can easily change answers in the generated questions if I find any of them not proper.	5-Step Likert Scale (Agreement)
I myself would create similar questions from these articles without this tool.	5-Step Likert Scale (Agreement)
What are your expectations from such a tool? What would you suggest?	Open

Table 2: Questionnaire for teachers.

The summarized evaluation results are as follows:

- The language learners find the exercise interface intuitive and suitable for a quick start on the exercises.
- Questions in the exercises are somewhat too easy to answer. Advanced learners are able to infer the correct answer without looking at the article text.
- The "hint" functionality is perceived ambiguously. While all users agree that such a functionality is helpful in principle, only some of the learners think the way it is implemented is helpful.
- Teachers find the multi-step exercise generation confusing, however, after one or two attempts they get familiar with it and can conveniently filter relation instances and modify answers.

- Generally, teachers think this is the kind of exercise they would create based on the given articles.

The results from the learner interviews indicate several problematic aspects of the prototype. Besides a usability issue with insufficient feedback during exercise conduction, an aspect mentioned frequently by the users relates to the complexity of the generated exercises. Since our testers are mainly advanced English learners, the exercises were relatively easy to solve for them. Apart from their English skill level, they also mentioned that answers with incorrect or less plausible gender statements are easy to exclude. For example an answer "*Guy Richtie gave birth to Loudres.*" is obviously false. We believe that such problems can be fixed by few, relation-specific heuristics, i.e., stricter rules on patterns. Another reason why questions tend to be easy is that the topic of the articles in our test set is celebrity gossip, i.e., an area which many people are familiar with, hence learners could answer questions based only on their prior knowledge, not their understanding of the text.

As for the tool for teachers, in the future we will provide clearer instructions so that teachers will not get lost in the process of creating exercises. According to the teachers, although some articles contain rich amounts of relations, they are not a good fit for a reading-comprehension exercise because of other aspects of the text. They also reported that at times none of the suggested answer options was acceptable; to solve this issue, we will add the option to freely edit the provided answer candidates, including the chance to compose totally new ones. In sum, the interview feedback from the teachers shows that despite the need for manual supervision, the overall prototype is perceived in a positive way.

5 Related work

The work presented in this paper is part of a growing body of approaches in computer-assisted language learning (CALL). The methods in this area aim to support (second) language learners through various means, among them methods for error correction (e.g., pronunciation training) or providing them with exercises for practicing existing language skills, while some approaches focus on reducing the workload of language teachers related to preparation and verification of exercises.

An example from the area of text-based CALL is the work of Uitdenbogerd (2014), who present

systems for finding or generating exercise texts of a complexity level appropriate to the learner's current skill level, e.g., by reordering existing text elements wrt. difficulty or finding texts which make use of only appropriate vocabulary. Similar work is reported by Sheehan et al. (2014), who classify texts wrt. different metrics (academic vocabulary, syntactic complexity, concreteness, cohesion, among others) in order to identify texts for specific complexity levels.

An area receiving particular focus in the literature is the task of reading comprehension. Typically, language learners are asked to provide a short free-text summary for, e.g., a news article. A teacher then has to manually verify whether the learner was capable of understanding the text and correctly summarized the main content. Some CALL systems support the teacher in this task by automatically scoring the learner's summary wrt. the original article text or compared to a teacher-provided gold-standard summary, see for example (Hahn and Meurers, 2012; Madnani et al., 2013; Horbach et al., 2013; Koleva et al., 2014).

Equally relevant to our work is the approach of Gates (2008), who automatically generated WH-questions for reading-comprehension tests through a transformation of the parse tree of selected sentences from the article text, as well as Riloff and Thelen (2000), who developed a rule-based system for the automatic answering of questions in a reading-comprehension setting.

Our focus is the automatic generation of multiple-choice reading-comprehension exercises. This exercise type is a standard tool for educational tests and has, compared to short-answer summaries, the benefit that once created such tests require relatively few work on the teacher's side in order to assess a learner's skill level. At the core of our approach is the application of existing information-extraction approaches, mainly from the sub-area of relation extraction, for the identification of facts in texts which are suitable for checking a learner's understanding of a foreign language. In addition to the work of Moro et al. (2013), which we employed in our prototype implementation, many more relation extraction systems exist that could be utilized in our setting, either from traditional relation extraction (Carlson et al., 2010; Mitchell et al., 2015) or the open-IE paradigm (Fader et al., 2011; Pighin et al., 2014).

6 Conclusion and outlook

In this paper, we present a semi-automatic approach to the generation of reading-comprehension exercises, which builds on existing strategies from the areas of information extraction and paraphrasing. A user evaluation of a prototype implementation provided some evidence for the feasibility of the approach, albeit it also showed that the quality and particularly the difficulty of the generated questions needs to improve.

For the future, we plan to implement further prototypes which will employ additional relation-extraction and paraphrasing systems, and which will support a broader range of fact types. Furthermore, we want to enlarge the lexical and syntactic variability of the generated answers and would like to reduce the amount of required teacher supervision, in order to make the approach better suited for real-world applications.

Another line of future work could focus on injecting more indirectness into the question-answer generation, i.e., the system should not only ask for facts explicitly referenced in the text but should also check a language learners conclusion capabilities, which require a deeper understanding of language than fact finding. A possible way to implement this may be the integration of textual-entailment methods. For example, the system might ask about a particular *parent-child* relation not directly mentioned in the text, which the system could infer from a mentioned relation between *siblings* and another (different) instance of relation *parent-child*. This can help to generate exercises testing for more sophisticated reading-comprehension capabilities of language learners.

Acknowledgments

This research was partially supported by the German Federal Ministry of Education and Research (BMBF) through the projects ALL SIDES (01IW14002) and Deependance (01IW11003).

References

- Renlong Ai, Marcela Charfuelan, Walter Kasper, Tina Klüwer, Hans Uszkoreit, Feiyu Xu, Sandra Gasber, and Philip Gienandt. 2014. Sprinter: Language technologies for interactive and multimedia language learning. In *LREC*.
- Andrew Carlson, Justin Betteridge, Bryan Kisiel, Burr Settles, Estevam R. Hruschka, and Tom M. Mitchell.

2010. Toward an Architecture for Never-Ending Language Learning. In *Twenty-Fourth AAAI Conference on Artificial Intelligence*, pages 1306–1313.
- Anthony Fader, Stephen Soderland, and Oren Etzioni. 2011. Identifying Relations for Open Information Extraction. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing, EMNLP '11*, pages 1535–1545, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Jenny Rose Finkel, Trond Grenager, and Christopher Manning. 2005. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 363–370, Ann Arbor, Michigan, June. Association for Computational Linguistics.
- Donna M. Gates. 2008. Automatically generating reading comprehension look-back strategy: Questions from expository texts. Master's thesis, Carnegie Mellon University.
- Michael Hahn and Detmar Meurers. 2012. Evaluating the meaning of answers to reading comprehension questions: A semantics-based approach. In *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP (BEA)*, pages 326–336, Montréal, Canada, June. Association for Computational Linguistics.
- Andrea Horbach, Alexis Palmer, and Manfred Pinkal. 2013. Using the text to evaluate short answers for reading comprehension exercises. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 1: Proceedings of the Main Conference and the Shared Task: Semantic Textual Similarity*, pages 286–295, Atlanta, Georgia, USA, June. Association for Computational Linguistics.
- Nikolina Koleva, Andrea Horbach, Alexis Palmer, Simon Ostermann, and Manfred Pinkal. 2014. Paraphrase detection for short answer scoring. In *Proceedings of the third workshop on NLP for computer-assisted language learning (NLP4CALL)*.
- Sebastian Krause, Hong Li, Feiyu Xu, Hans Uszkoreit, Robert Hummel, and Luise Spielhagen. 2014. Language resources and annotation tools for cross-sentence relation extraction. In *LREC*.
- Nitin Madnani, Jill Burstein, John Sabatini, and Tenaha O'Reilly. 2013. Automated scoring of a summary-writing task designed to measure reading comprehension. In *Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications (BEA)*, pages 163–168, Atlanta, Georgia, June. Association for Computational Linguistics.
- T. Mitchell, W. Cohen, E. Hruschka, P. Talukdar, J. Betteridge, A. Carlson, B. Dalvi, M. Gardner, B. Kisiel, J. Krishnamurthy, N. Lao, K. Mazaitis, T. Mohamed, N. Nakashole, E. Platanios, A. Ritter, M. Samadi, B. Settles, R. Wang, D. Wijaya, A. Gupta, X. Chen, A. Saporov, M. Greaves, and J. Welling. 2015. Never-ending learning. In *Proceedings of the Conference on Artificial Intelligence (AAAI)*.
- Andrea Moro, Hong Li, Sebastian Krause, Feiyu Xu, Roberto Navigli, and Hans Uszkoreit. 2013. Semantic rule filtering for web-scale relation extraction. In *ISWC*.
- Ndapandula Nakashole, Gerhard Weikum, and Fabian M. Suchanek. 2012. PATTY: A taxonomy of relational patterns with semantic types. In Jun'ichi Tsujii, James Henderson, and Marius Pasca, editors, *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, EMNLP-CoNLL 2012, July 12-14, 2012, Jeju Island, Korea*.
- Daniele Pighin, Marco Cornolti, Enrique Alfonseca, and Katja Filippova. 2014. Modelling events through memory-based, open-ie patterns for abstractive summarization. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, Volume 1: Long Papers*, pages 892–901, Baltimore, MD, USA, June.
- Ellen Riloff and Michael Thelen. 2000. A rule-based question answering system for reading comprehension tests. In *Proceedings of the 2000 ANLP/NAACL Workshop on Reading Comprehension Tests As Evaluation for Computer-based Language Understanding Systems - Volume 6, ANLP/NAACL-ReadingComp '00*, pages 13–19, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Kathleen M. Sheehan, Irene Kostin, Diane Napolitano, and Michael Flor. 2014. The TextEvaluator tool: Helping teachers and test developers select texts for use in instruction and assessment. *The Elementary School Journal*, 115(2):184–209, December.
- Alexandra Uitdenbogerd. 2014. Tools for supporting language acquisition via extensive reading. In *Proceedings of the 1st Workshop on Natural Language Processing Techniques for Educational Applications (NLP-TEA-1) at the 22nd International Conference in Computer Education (ICCE 2014)*.
- Rui Wang and Yi Zhang. 2012. Sentence realization with unlexicalized tree linearization grammars. In *Proceedings of COLING 2012: Posters*, Mumbai, India, December.
- Feiyu Xu, Hans Uszkoreit, and Hong Li. 2007. A seed-driven bottom-up machine learning framework for extracting relations of various complexity. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, page 584–591, Prague, Czech Republic, June. Association for Computational Linguistics.