A Sentence Simplification System for Improving Relation Extraction

Christina Niklaus, Bernhard Bermeitinger, Siegfried Handschuh, André Freitas
Faculty of Computer Science and Mathematics
University of Passau
Innstr. 41, 94032 Passau, Germany
{christina.niklaus, bernhard.bermeitinger, siegfried.handschuh, andre.freitas }@uni-passau.de

Abstract

In this demo paper, we present a text simplification approach that is directed at improving the performance of state-of-the-art Open Relation Extraction (RE) systems. As syntactically complex sentences often pose a challenge for current Open RE approaches, we have developed a simplification framework that performs a pre-processing step by taking a single sentence as input and using a set of syntactic-based transformation rules to create a textual input that is easier to process for subsequently applied Open RE systems.

1 Introduction

Relation Extraction (RE) is the task of recognizing the assertion of relationships between two or more entities in NL text. Traditional RE systems have concentrated on identifying and extracting relations of interest by taking as input the target relations, along with hand-crafted extraction patterns or patterns learned from hand-labeled training examples. Consequently, shifting to a new domain requires to first specify the target relations and then to manually create new extraction rules or to annotate new training examples by hand (Banko and Etzioni, 2008). As this manual labor scales linearly with the number of target relations, this supervised approach does not scale to large, heterogeneous corpora which are likely to contain a variety of unanticipated relations (Schmidek and Barbosa, 2014). To tackle this issue, Banko and Etzioni (2008) introduced a new extraction paradigm named 'Open RE' that facilitates domain-independent discovery of relations extracted from text by not depending on any relation-specific human input.

Generally, state-of-the-art Open RE systems identify relationships between entities in a sentence by matching patterns over either its POS tags, e. g. (Banko et al., 2007; Fader et al., 2011; Merhav et al., 2012), or its dependency tree, e. g. (Nakashole et al., 2012; Mausam et al., 2012; Xu et al., 2013; Mesquita et al., 2013). However, particularly in long and syntactically complex sentences, relevant relations often span several clauses or are presented in a non-canonical form (Angeli et al., 2015), thus posing a challenge for current Open RE approaches which are prone to make incorrect extractions - while missing others - when operating on sentences with an intricate structure.

To achieve a higher accuracy on Open RE tasks, we have developed a framework for simplifying the linguistic structure of NL sentences. It identifies components of a sentence which usually provide supplementary information that may be easily extracted without losing essential information. By applying a set of hand-crafted grammar rules that have been defined in the course of a rule engineering process based on linguistic features, these constituents are then disembedded and transformed into self-contained simpler context sentences. In this way, sentences that present a complex syntax are converted into a set of more concise sentences that are easier to process for subsequently applied Open RE systems, while still expressing the original meaning.

Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: System Demonstrations, pages 170–174, Osaka, Japan, December 11-17 2016.

2 System Description

Referring to previous attempts at syntax-based sentence compression (Dunlavy et al., 2003; Zajic et al., 2007; Perera and Kosseim, 2013), the idea of our text simplification framework is to syntactically simplify a complex input sentence by splitting conjoined clauses into separate sentences and by eliminating specific syntactic sub-structures, namely those containing only minor information. However, unlike recent approaches in the field of extractive sentence compression, we do not delete these constituents, which would result in a loss of background information, but rather aim at preserving the full informational content of the original sentence. Thus, on the basis of syntax-driven heuristics, components which typically provide mere secondary information are identified and transformed into simpler stand-alone context sentences with the help of paraphrasing operations adopted from the text simplification area.

Definition of the Simplification Rules By analyzing the structure of hundreds of sample sentences from the English Wikipedia, we have determined constituents that commonly supply no more than contextual background information. These components comprise the following syntactic elements:

- **non-restrictive relative clauses** (e. g. *"The city's top tourist attraction was the Notre Dame Cathedral, which welcomed 14 million visitors in 2013."*)
- **non-restrictive** (e. g. *"He plays basketball, a sport he participated in as a member of his high school's varsity team."*) **and restrictive appositive phrases** (e. g. *"He met with former British Prime Minister Tony Blair."*)
- **participial phrases offset by commas** (e. g. *"The deal, titled Joint Comprehensive Plan of Action, saw the removal of sanctions."*)
- **adjective and adverb phrases delimited by punctuation** (e. g. *"Overall, the economy expanded at a rate of 2.9 percent in 2010."*)
- **particular prepositional phrases** (e. g. *"In 2012, Time magazine named Obama as its Person of the Year."*)
- **lead noun phrases** (e. g. *"Six weeks later, Alan Keyes accepted the Republican nomination."*)
- **intra-sentential attributions** (e. g. *"He said that both movements seek to bring justice and equal rights to historically persecuted peoples."*)
- **parentheticals** (e. g. *"He signed the reauthorization of the State Children's Health Insurance Program (SCHIP)."*)

Besides, both conjoined clauses presenting specific features and sentences incorporating particular punctuation are disconnected into separate ones.
After having thus identified syntactic phenomena that generally require simplification, we have determined the characteristics of those constituents, using a number of syntactic features (constituency-based parse trees as well as POS tags) that have occasionally been enhanced with the semantic feature of NE tag. For computing them, a number of software tools provided by the Stanford CoreNLP framework have been employed (Stanford Parser, Stanford POS Tagger and Stanford Named Entity Recognizer).[1] Based upon these properties, we have then specified a set of hand-crafted grammar rules for carrying out the syntactic simplification operations which are applied one after another on the given input sentence. In that way, linguistically peripheral material is disembedded, thus producing a more concise core sentence which is augmented by a number of related self-contained contextual sentences (see the example displayed in figure 1).

[1] http://nlp.stanford.edu/software/

Algorithm 1 Syntax-based sentence simplification

Input: sentence s
repeat
  $r \leftarrow$ next rule $\in$ R # Null if no more rules
  if r is applicable to s **then**
    $C, P \leftarrow$ apply $r_{extract}$ to s # Identify the set of constituents C to extract from s, and their positions P in s
    for all constituents $c \in C$ **do**
      $context \leftarrow$ apply $r_{paraphrase}$ to c # Produce a context sentence
      $contextSet \leftarrow$ add $context$ # Add it to the core's set of associated context sentences
until $R = \emptyset$
$core \leftarrow$ delete tokens in s at positions $p \in P$ # Reduce the input to its core
return $core$ and $contextSet$ # Output the core and its context sentences

Application of the Simplification Operations The simplification rules we have specified are applied one after another to the source sentence, following a three-stage approach (see algorithm 1). First, clauses or phrases that are to be separated out - including their respective antecedent, where required - have to be identified by pattern matching. In case of success, a context sentence is constructed by either linking the extractable component to its antecedent or by inserting a complementary constituent that is required in order to make it a full sentence. Finally, the main sentence has to be reduced by dropping the clause or phrase, respectively, that has been transformed into a stand-alone context sentence.

In this way, a complex source sentence is transformed into a simplified two-layered representation in the form of core facts and accompanying contexts, thus providing a kind of normalization of the input text. Accordingly, when carrying out the task of extracting semantic relations between entities on the reduced core sentences, the complexity of determining intricate predicate-argument structures with variable arity and nested structures from syntactically complex input sentences is removed. Beyond that, the phrases of the original sentence that convey no more than peripheral information are converted into independent sentences which, too, can be more easily extracted under a binary or ternary predicate-argument structure (see the example illustrated in figure 1).

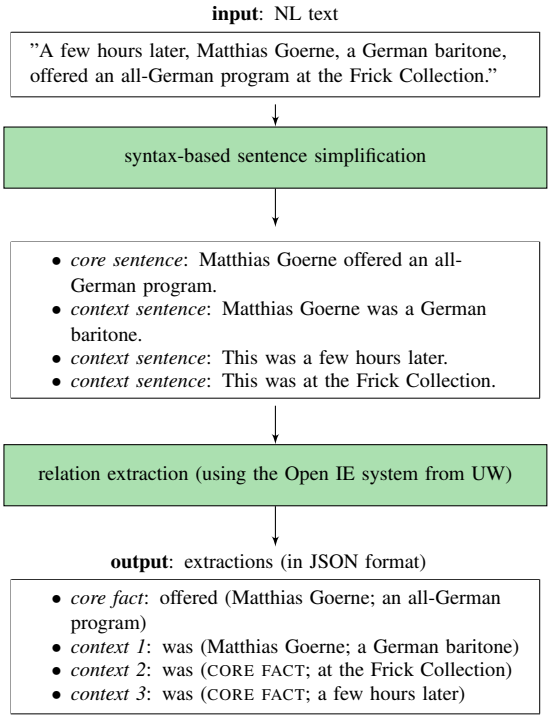

Figure 1: Simplification and extraction pipeline

3 Evaluation

The results of an experimental evaluation show that state-of-the-art Open RE approaches obtain a higher accuracy and lower information loss when operating on sentences that have been pre-processed by our simplification framework. In particular when dealing with sentences that contain nested structures, Open RE systems benefit from a prior simplification step (see figures 2 and 3 for an example). The full evaluation methodology and detailed results are reported in Niklaus et al. (2016).

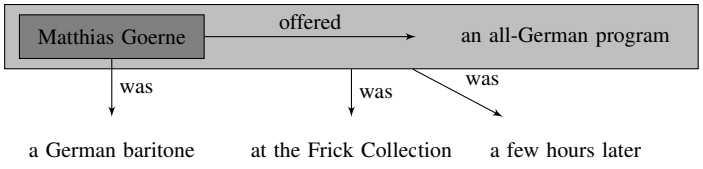

Figure 2: Extracted relations when operating on the simplified sentences

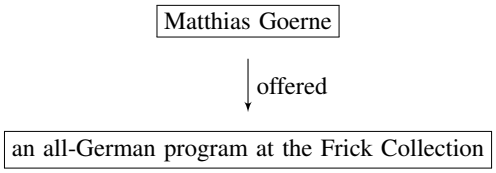

Figure 3: Result without a prior simplification step

4 Usage

The text simplification framework is publicly available[2] as both a library and a command line tool whose workflow is depicted in figure 1. It takes as input NL text in the form of either a single sentence or a file with line separated sentences. As described above, each input sentence is first transformed into a structurally simplified version consisting of 1 to n core sentences and 0 to m associated context sentences. In a second step, the relations contained in the input are extracted by applying the Open IE system[3] upon the simplified sentences. Finally, the results generated in this way are written to the console or a specified output file in JSON format. As an example, the outcome produced by our simplification system when applied to a full Wikipedia article is provided online.[4]

5 Conclusion

We have described a syntax-driven rule-based text simplification framework that simplifies the linguistic structure of input sentences with the objective of improving the coverage of state-of-the-art Open RE systems. As an experimental analysis has shown, the text simplification pre-processing improves the result of current Open RE approaches, leading to a *lower information loss* and a *higher accuracy* of the extracted relations.

References

Gabor Angeli, Melvin Johnson Premkumar, and Christopher D. Manning. 2015. Leveraging linguistic structure for open domain information extraction. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pages 344–354. ACL.

Michele Banko and Oren Etzioni. 2008. The tradeoffs between open and traditional relation extraction. In *Proceedings of ACL-08: HLT*, pages 28–36. ACL.

Michele Banko, Michael J. Cafarella, Stephen Soderland, Matt Broadhead, and Oren Etzioni. 2007. Open information extraction from the web. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence*, pages 2670–2676.

Daniel M. Dunlavy, John M. Conroy, Judith D. Schlesinger, Sarah A. Goodman, Mary Ellen Okurowski, Dianne P. O'Leary, and Hans van Halteren. 2003. Performance of a three-stage system for multi-document summarization.

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*, pages 1535–1545. ACL.

Mausam, Michael Schmitz, Robert Bart, Stephen Soderland, and Oren Etzioni. 2012. Open language learning for information extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 523–534. ACL.

Yuval Merhav, Filipe Mesquita, Denilson Barbosa, Wai Gen Yee, and Ophir Frieder. 2012. Extracting information networks from the blogosphere. *ACM Trans. Web*, 6(3):11:1–11:33.

[2] `https://gitlab.com/nlp-passau/SimpleGraphene`
[3] `https://github.com/allenai/openie-standalone`
[4] `https://gitlab.com/nlp-passau/SimpleGraphene/tree/master/examples`

Filipe Mesquita, Jordan Schmidek, and Denilson Barbosa. 2013. Effectiveness and efficiency of open relation extraction. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 447–457. ACL.

Ndapandula Nakashole, Gerhard Weikum, and Fabian Suchanek. 2012. Patty: A taxonomy of relational patterns with semantic types. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1135–1145. ACL.

Christina Niklaus, Siegfried Handschuh, and André Freitas. 2016. Improving relation extraction by syntax-based sentence simplification. `https://gitlab.com/nlp-passau/SimpleGraphene/blob/master/paper/improving-relation-extraction.pdf`.

Prasad Perera and Leila Kosseim. 2013. Evaluating syntactic sentence compression for text summarisation. In *Natural Language Processing and Information Systems - 18th International Conference on Applications of Natural Language to Information Systems*, pages 126–139.

Jordan Schmidek and Denilson Barbosa. 2014. Improving open relation extraction via sentence re-structuring. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation*. ELRA.

Ying Xu, Mi-Young Kim, Kevin Quinn, Randy Goebel, and Denilson Barbosa. 2013. Open information extraction with tree kernels. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 868–877. ACL.

David Zajic, Bonnie J. Dorr, Jimmy Lin, and Richard Schwartz. 2007. Multi-candidate reduction: Sentence compression as a tool for document summarization tasks. *Information Processing and Management*, 43(6):1549–1570.