From Strings to Things
SAR-Graphs: A New Type of Resource for Connecting Knowledge and Language

Hans Uszkoreit and Feiyu Xu

Language Technology Lab, DFKI, Alt-Moabit 91c, Berlin, Germany
{uszkoreit,feiyu}@dfki.de

Abstract. Recent research and development have created the necessary ingredients for a major push in web-scale language understanding: large repositories of structured knowledge (DBpedia, the Google knowledge graph, Freebase, YAGO) progress in language processing (parsing, information extraction, computational semantics), linguistic knowledge resources (Treebanks, WordNet, BabelNet, UWN) and new powerful techniques for machine learning. A major goal is the automatic aggregation of knowledge from textual data. A central component of this endeavor is relation extraction (RE). In this paper, we will outline a new approach to connecting repositories of world knowledge with linguistic knowledge (syntactic and lexical semantics) via web-scale relation extraction technologies.

Keywords: Knowledge Graph, Grammar-based Relation Extraction Rules, Relation-specific lexical semantic graphs, Linking linguistic resources

1 Motivation

The powerful vision of a semantic web has already started materializing through large repositories of structured knowledge extracted from Wikipedia and other sources of texts or data bases. The existence of these knowledge resources and further progress in linking open data collections has nurtured the demand for even more extensive sources of structured knowledge. Although some (linked) open data collections can feed into the stock of digital knowledge, the most effective growth is expected from automatic extraction of information and knowledge out of texts. Knowledge repositories such as the DBpedia [1] are not only driving advanced research in this important field but they also serve as important resources for the applied learning methods. Relation extraction with distantly supervised learning (e.g., [5, 6]) utilizes the facts in Freebase [2], DBpedia, or Yago [4] as seed knowledge for the discovery of the relevant extraction patterns in large volumes of texts or even on the entire indexed web.

In our own research [5, 7, 9–12], we were able to train relation extraction for n-ary relations with the help of examples of facts or events, e.g., with hundreds of thousands of sample facts borrowed from Freebase. For each of the n-ary relations

pieced together from Freebase facts, we automatically learned around hundred thousand extraction rules that work on the dependency structures of searched texts. With these rules we achieved a higher recall on detecting relation instances in unseen texts than any of our previous methods. Unfortunately, these rule sets are rather noisy: The majority of the learned rules are not appropriate for accurate extraction, therefore the method yields very low precision. However, we found ways to filter the acquired rules semantically. Some of these filters exploit the knowledge of Freebase, newer ones also utilize another type of knowledge repositories, i.e., lexical semantic networks such as WordNet [3] and BabelNet [8]. With these filtering techniques, we were able to boost precision [7].

However, so far the learned statistical models or rule/pattern sets are not freely usable. For any given relation (including events and facts) in DBpedia, Freebase or Yago, we have many instances but we do not have a knowledge resource that tells us for the covered types of relations and facts which patterns or lexical concepts a language owns to represent and describe instances of these relations. If we had such a resource, it would be comparatively easy to build extraction engines for any of the relations. We are happy to share our extraction rules but they come in a special format suited for our relation extraction system DARE [9, 12]. Even if other researchers could use the rules by transforming them to their preferred format, it would remain unlikely that the rule set will eventually become a widely shared resource collectively maintained and actively extended by many research groups.

Instead of trying to promote our rule set and format, we would like to propose in this paper a new knowledge resource that for each covered target relation contains dependency structures of linguistic constructions representing the target relation itself and semantically associated relations, which also indicate an instance of the target relation. We call this new resource type a **SAR**-graph, a dependency graph of **S**emantically **A**ssociated **R**elations. We could also term such a graph a *language graph*, because it represents the linguistic patterns for a relation in a knowledge graph. A language graph can be thought of as a bridge between the language and the knowledge graph, a bridge that characterizes the ways in which a language can express instances of a relation, and thus a mapping from strings to things. First samples of such language graphs have been built by means of the rule learning facility of the DARE relation extraction system. But we will also propose a simple and straightforward instrument for populating these language graphs by annotated examples. These are sentences containing a mention of a relation instance.

2 Automatic Acquisition of Relation Extraction Rules

DARE can handle target relations of varying arity through a compositional and recursive rule representation and a bottom-up rule discovery strategy. A DARE rule for an n-ary relation can be composed of rules for its projections, namely, rules that extract a subset of the n arguments. Furthermore, it defines explicitly the semantic roles of linguistic arguments for the target relation. The following

examples illustrate the DARE rule and its extraction strategy. *Example 1.* is a relation instance of the target relation from [9] concerning Prize awarding event, which contains four arguments: *Winner*, *Prize_ Name*, *Prize_ Area* and *Year*. *Example 1.* refers to an event mentioned in *Example 2.*

Example 1. <*Mohamed ElBaradei, Nobel, Peace, 2005*>.

Example 2. *Mohamed ElBaradei*, won the *2005 Nobel* Prize for *Peace* on Friday.

Given *Example 1.* as a seed, *Example 1.* matches with the sentence in *Example 2.* and DARE assigns the semantic roles known in the seed to the matched linguistic arguments in *Example 2.* Fig. 1. is a simplified dependency tree of *Example 2.* with named entity annotations and corresponding semantic role labelling after the match with the seed. DARE utilizes a bottom-up rule discovery strategy to extract rules from such semantic role labelled dependency trees.

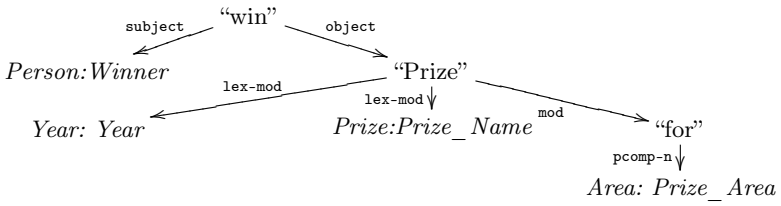

Fig. 1: Dependency tree of *Example 2.* matched with the seed

Rule name :: winner_prize_area_year_1
Rule body ::

$$\left[\begin{array}{ll} \text{head} & \left[\begin{array}{ll} \text{pos} & \text{verb} \\ \text{mode} & \text{active} \\ \text{lex-form} & \text{“win”} \end{array}\right] \\ \text{daughters} & < \left[\text{subject} \quad \left[\text{head} \quad \boxed{1}\textit{Person}\right]\right], \\ & \left[\text{object} \quad \left[\begin{array}{ll} \text{rule} & \text{year\_prize\_area\_1} :: \\ & < \boxed{4}\textit{Year}, \boxed{2}\textit{Prize\_ Name}, \\ & \boxed{3}\textit{Prize\_ Area} > \end{array}\right]\right] > \end{array}\right]$$

Output :: < $\boxed{1}$*Winner*, $\boxed{2}$*Prize_ Name*, $\boxed{3}$*Prize_ Area*, $\boxed{4}$*Year* >

Fig. 2: DARE extraction rule.

From the tree in Fig. 1., DARE learns three rules in a bottom-up manner, each step with a one tree depth. The first rule is extracted from the subtree dominated by the preposition "for", extracting the argument *Prize_ Area* (*Area*), while the second rule makes use of the subtree dominated by the noun "Prize", extracting the arguments *Year* (*Year*) and *Prize_ Name* (*Prize*), and calling the first rule for the argument *Prize_ Area* (*Area*). The third rule "winner_prize_area_year_1" is depicted in Figure 2. The value of *Rule body* is

extracted from the dependency tree. In "winner_prize_area_year_1", the subject value *Person* fills the semantic role *Winner*. The object value calls internally the second rule called "year_prize_area_1", which handles the other arguments *Year* (*Year*), *Prize_Name* (*Prize*) and *Prize_Area* (*Area*).

The Web-DARE system is a further development of the DARE system [5]. Web-DARE learns RE rules for n-ary relations in a distant-supervision manner [6], namely, utilization of a large amount of seed examples with only one iteration step, no bootstrapping involved. For 39 relations, 200k instances, i. e. *seeds*, were collected from the freely-available knowledge base Freebase. Utilizing these relation instances as Web-search queries, a total of 20M Web pages was retrieved and processed, extracting from them 3M sentences mentioning the arguments (entities) of a seed instance. After analyzing these sentences by additional NER and parsing, 1.5M RE rules were extracted from the dependency parses.

3 Linking Knowledge Graphs with Language

We have started to use our learned extraction patterns as start content for building an open resource that bridges the gap between the world knowledge as encoded in knowledge repositories on the one hand and the representation of facts and events in human language texts on the other. For each considered target relation represented in the knowledge graphs, such a resource will consist of merged dependency graphs for all relevant patterns learned from mentions of relation instances.

We will employ lexical knowledge bases such as BabelNet, WordNet, VerbNet, UWN to extend the content words in the dependency relations by semantically related words (synonyms, hyponyms, hyperonyms). This will allow further merging of subgraphs and will also make the sar-graphs more general. Such a sar-graph will help to identify and compose mentions of argument entities and projections of an n-ary relation. In Figure 3, we depict linkings between knowledge graphs, relation-specific dependency grammar based relation extraction rules and relation-specific lexical semantic graphs. Given the facts in the knowledge graphs such as Freebase and the free texts provided on WWW, relation extraction systems such as our DARE and Web-DARE systems can learn grammar-based relation extraction rules for each relation type available in the knowledge graphs. Given the relation extraction rules, sentence mentions from which the rules are learned and general lexical semantic network such as BabelNet, we can learn and extract relation-specific lexical semantic graphs as we have done in [7]. All three resources can be linked since they are about the same relation types, but from world knowledge or linguistic knowledge points of view.

4 SAR-Graphs

A sar-graph can be built for every n-ary relation $R\langle a_1, ..., a_n\rangle$ such as marriage $R\langle Person\_1, Person\_2, CeremonyLoc, FromDate, ToDate\rangle$ and every language l. A sar-graph of a relation R is a directed graph with labeled edges and vertices.

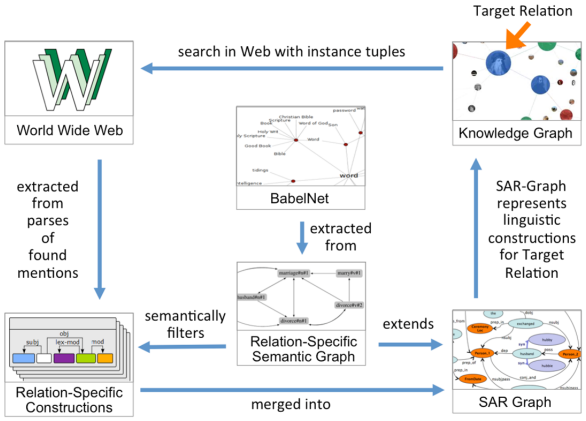

Fig. 3: Linking Knowledge Graph with Linguistic Resources

The relation R we call the target relation. The function of the sar-graph is to represent the linguistic constructions the language l provides for reporting instances of R or for just referring to such instances. The linguistic constructions are represented as dependency structures that only include words belonging to the construction and slots for the arguments. Thus a sar-graph is composed of syntactic dependency graphs. Their edges denote dependency relations. Each edge is labeled with the tag the parser has assigned to the dependency. Vertices come in two flavors: One type of vertices denotes a regular node in a dependency structure, thus it is labeled with a word. Vertices of the second type represent the slots for the arguments of the target relation, instead of a word, they are labeled by the name of the argument, e.g. $Person\_1$.

If some given language l had only one single construction to express an instance of R then the dependency structure of this construction would be the entire sar-graph. But if the language offered alternatives to this construction, i.e. paraphrases, their dependency structures would also be entered into the sar-graph. They would be connected in such a way that all vertices labeled by the same argument name are merged.

Our rules do not just detect constructions that denote the entire relation but also many constructions referring to aspects or parts of the relation instance. As long as these constructions indicate an instance of the target relation, they are needed for high-recall relation extraction. One type of constructions that are not true paraphrases of the constructions exactly expressing the n-ary relation, denote instances of projections of R. A sar-graph for the following two English constructions would look as presented in Figure 4.

Constructions that refer to some part or aspect of the relation would normally be seen as sufficient evidence of an instance even if there could be contexts in which this implication is canceled.

Example 3. Joan and Edward exchanged rings in 2011.

Example 4. Joan and Edward exchanged rings during the rehearsal of the ceremony.

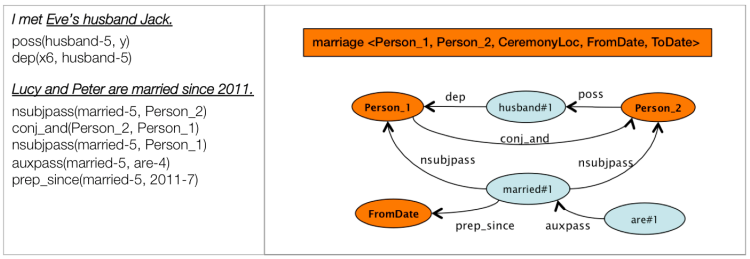

Fig. 4: Example of sar-graph for two English constructions

Other constructions refer to relations that entail the target relations without being part of it.

Example 5. Joan and Edward celebrated their 12th wedding anniversary.

Example 6. Joan and Edward got divorced in 2011.

And finally there are constructions referring to semantically connected relations that by themselves might not be used for safely detecting instances of R but that could be employed for recall-optimized applications or for the probabilistic detection process that combines several pieces of evidence.

Example 7. I met her last October at Joan's bachelorette(engagement) party.

Some entirely probabilistic entailments are caused by social conventions or behavioral preferences.

Example 8. Two years before Joan and Paul had their first child, they bought a larger home.

In a next step, we extend the sar-graphs by lexical semantic knowledge. During the semantic filtering of our rules [7], we disambiguated all content words in the rules. Therefore, we can mark the vertices by readings instead of head words. i.e., pairs of a word and its WordNet sense number. The same semantic knowledge base we used for disambiguation, also gives us readings of semantically related words for the word (actually readings) in our sar-graph. It also finds semantic relations among words already in the sar-graph. In order to add this additional information, we need to introduce a new type of edge for lexical semantic relations. These edges are labeled with semantic relation tags such as hypernym, synonym, troponym, or antonym. The synonyms, hyperonyms and troponyms that are not yet in the sar-graph will be added as new vertices.

The following artificially constrained example in Figure 5 may serve to illustrate the structure and contents of a sar-graph. The target relation is marriage again. We include only five constructions extracted from the five listed sentences. After each sentence we list the dependency relations from the full parse that belong to the construction. For better readability, we omit the reading numbers in this example.

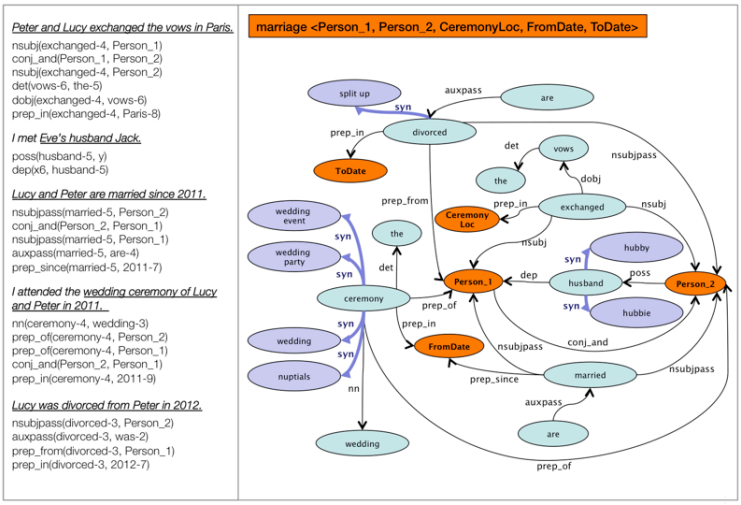

Fig. 5: Example of sar-graph including lexical semantic information

5 Applications

The most obvious application of our approach is information extraction. As the sar-graphs we have already built contain all the patterns we had learned and tested for n relations, we know that the information in a sar-graph can be successfully applied to regular relation extraction, i.e. the detection of instances in sentences. Without being able to already prove it, we also believe that sar-graphs are especially suited for the extraction of n-ary relations across sentence boundaries. When we worked on relation extraction from sequences of sentences [13] we discovered that coreference resolution does not always suffice to piece a relation instance together from several parts. In addition to the local rules for the parts, we need to know which role the detected participants have in the full n-ary relation instance. Assume the following text:

Example 9. My boss, Janice Miller, is married to a Texan physician. Her hubby Jack is a urologist from Dallas. I attended their wedding event in 2011.

Here several associated relations from the sar-graph are applied including the synonym *hubby* for *husband*. The sar-graph also tells us which roles of the target relation are filled by the associated relations.

We expect that sar-graphs could also become useful for summarization, since they permit to find constructions that express all or most arguments of a relation in one sentence. Finally, they might be employed for the generation of sentences from database facts, e.g., in reports. Because of the range of paraphrases, generation could produce stylistic variation as extensively used in reports written by human authors.

6 Populating sar-graphs

As described above, the first sar-graphs were simply built by combining the dependency structures in automatically acquired relation extraction rules. But

this is not the only way to populate sar-graphs. Just as the DARE rule-learning formalism can extract a rule from a mention, the extraction could also happen from a sample sentence. All that is needed is some annotation marking the arguments of the target relation. The sentence will then be parsed. The dependency structure of the construction will be determined as the minimal spanning tree containing the arguments. The instantiated arguments will be substituted by the names of the argument places given in the markup. The content words of the extracted structure will be disambiguated utilizing BabelNet. If the content words are not already contained in the sar-graph, semantically related content words will also be added as exemplified in Figure 5.

7 Conclusion

We have shown how to combine the automatically accumulated knowledge about the means a language provides for speaking about a certain relation into one connected graph. We have also described, how such a graph could be built or extended from annotated examples.

A network that combines several semantic relations describing different parts or aspects of a fragment of the world is somewhat reminiscent of so-called semantic nets. As such the semantic combination of multiple relations seems to belong into the semantic knowledge base, such as the knowledge graph or some specialized ontology. So why do we not simply build a language-independent semantic network first and then look for the linguistic constructions that different languages use to express the relations. Such a strategy would throw us back to a research paradigm in which knowledge engineering precedes any attempt of language understanding. From experience we have learned that there could be numerous different ontologies just for the thematic area *marriage*. Lawyers, event managers, relationship counselors, vital statisticians may come up with completely different ways to select and structure the relevant knowledge pieces. How could we decide on the best ontology for relation extraction? Would any of such intellectually created ontologies contain a relation for exchanging the vows and one for tying the knot? How would the vows and the knot be represented? The great advantage of the bottom up empirical approach we are taking is that our sar-graphs are determined by the way people refer to a relation (event type, process, etc.). This makes them suited for semantic text analytics including information extraction.

Another important advantage is the association of a graph to a specific language. A Greek report on a wedding may refer to the wedding crowns for bride and groom, in an English sar-graph for the marriage relation, such crowns would not show up. In a Greek wedding the betrothal can be a part of the entire ceremony in other cultures it must have taken place a certain period before the wedding. In some cultures, exchanging the rings means getting married in others there is no such concept.

We are convinced that we need the interaction of two strategies to build up a growing stock of structured knowledge in the spirit of a semantic web. One strat-

egy that starts from structuring growing portions of textual knowledge sources such as the Wikipedia and extends this knowledge by structured data such as linked open data, and another strategy that uses and extends the resulting repositories of structured knowledge by extracting from all sorts of texts much more knowledge, especially contingent knowledge. The novel type of repository we have proposed will on the one hand facilitate the latter process and on the other hand maintain the link of the accumulated domain-sorted linguistic knowledge with DBpedia, knowledge graph or similar knowledge resources.

Acknowledgements. This research was partially supported by the German Federal Ministry of Education and Research (BMBF) through the projects Deependance (contract 01IW11003), by Google through a Faculty Research Award granted in July 2012 and a Focused Research Award granted in July 2013.

References

1. Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., Hellmann, S.: {DBpedia} - a crystallization point for the web of data. Web Semantics: Science, Services and Agents on the World Wide Web **7**(3) (2009) 154 – 165
2. Bollacker, K.D., Evans, C., Paritosh, P., Sturge, T., Taylor, J.: Freebase: a collaboratively created graph database for structuring human knowledge. In: Proc. of SIGMOD. (2008) 1247–1250
3. Fellbaum, C., ed.: WordNet: an electronic lexical database. Christiane Fellbaum, Cambridge, MA, USA (1998)
4. Hoffart, J., Suchanek, F.M., Berberich, K., Weikum, G.: YAGO2: A spatially and temporally enhanced knowledge base from Wikipedia. Artificial Intelligence **194** (2013) 28–61
5. Krause, S., Li, H., Uszkoreit, H., Xu, F.: Large-scale learning of relation-extraction rules with distant supervision from the web. In: Proc. of 11th ISWC, Part I. (2012) 263–278
6. Mintz, M., Bills, S., Snow, R., Jurafsky, D.: Distant supervision for relation extraction without labeled data. In: Proc. of ACL/AFNLP. (2009) 1003–1011
7. Moro, A., Li, H., Krause, S., Xu, F., Navigli, R., Uszkoreit, H.: Semantic rule filtering for web-scale relation extraction. In: Proceeding of International Semantic Web Conference (ISWC 2013), to appear. (2013)
8. Navigli, R., Ponzetto, S.P.: BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. Artificial Intelligence **193** (2012) 217–250
9. Xu, F.: Bootstrapping Relation Extraction from Semantic Seeds. PhD thesis, Saarland University (2007)
10. Xu, F., Li, H., Zhang, Y., Uszkoreit, H., Krause, S.: Parse reranking for domain-adaptative relation extraction. Journal of Logic and Computation (2012)
11. Xu, F., Uszkoreit, H., Krause, S., Li, H.: Boosting relation extraction with limited closed-world knowledge. In: Proc. of COLING (Posters). (2010) 1354–1362
12. Xu, F., Uszkoreit, H., Li, H.: A seed-driven bottom-up machine learning framework for extracting relations of various complexity. In: Proc. of ACL. (2007)
13. Xu, F., Uszkoreit, H., Li, H.: Task driven coreference resolution for relation extraction. In: Proceedings of the European Conference for Artificial Intelligence ECAI 2008, Patras, Greece (2008)