REFRACTIVE: An Open Source Tool to Extract Knowledge from Syntactic and Semantic Relations

Peter Exner, Pierre Nugues
Lund University, Department of Computer science, Lund, Sweden
peter.exner@cs.lth.se, pierre.nugues@cs.lth.se

Abstract

The extraction of semantic propositions has proven instrumental in applications like IBM Watson (Ferrucci, 2012) and in Google's knowledge graph (Singhal, 2012). One of the core components of IBM Watson is the PRISMATIC knowledge base consisting of one billion propositions extracted from the English version of Wikipedia and the *New York Times* (Fan et al., 2010). However, extracting the propositions from the English version of Wikipedia is a time-consuming process. In practice, this task requires multiple machines and a computation distribution involving a good deal of system technicalities. In this paper, we describe REFRACTIVE, an open-source tool to extract propositions from a parsed corpus based on the Hadoop variant of MapReduce. While the complete process consists of a parsing part and an extraction part, we focus here on the extraction from the parsed corpus and we hope this tool will help computational linguists speed up the development of applications.

Keywords: semantic parsing, relation extraction, proposition database.

1. Introduction

IBM Watson (Ferrucci, 2012) and Google's knowledge graph (Singhal, 2012) are recent and spectacular achievements that show the significance of large scale semantic processing. Systems like IBM Watson and OLLIE (Mausam et al., 2012) have carried out a systematic extraction of semantic frames on very large corpora. Both systems used a syntactically parsed corpus as input, grammatical relations and rules to derive the predicate-argument structures as it is fast and easier to apply to large corpora. A complete semantic role labeling, using the PropBank (Palmer et al., 2005) or Framenet (Ruppenhofer et al., 2010) lexicons, is slower, but usually more accurate. Nonetheless, it is possible to combine both techniques to get higher performances. See Mausam et al. (2012) for a discussion.

While distributing the extraction manually on two or more computers seems an intuitive solution, it is difficult to implement in practice. Applying a semantic parser to corpora of more than 1 billion words is a daunting task that requires a computer cluster to be tractable in practice. On average, the end-to-end Mate semantic pipeline (Björkelund et al., 2010) takes 190 milliseconds to parse a sentence with a high-end single CPU. Given that the English version has 62 million sentences, the corpus parsing would be completed in ideal conditions in little less than five months. Such a duration does not fit the interactive, try-and-fail nature of many experiments in natural language processing.

In this paper, we introduce REFRACTIVE, a streamlined, large-scale extraction tool based on MapReduce that runs on a computer cluster of arbitrary size. Using it, we extracted propositions from a parsed version of the entire English Wikipedia and we created a knowledge base from it. The complete processing, parsing and extraction, takes less than 20 days on a cluster of ten processors, where we represent the extracted propositions as frames and slots. In the rest of the paper, we describe the extraction part of the process.

2. Related Work

Lexical databases or networks representing structured general knowledge have a long history in natural language processing. Examples resorting to an intensive manual labor include WordNet (Miller and Fellbaum, 1998), FrameNet, and Cyc (Lenat, 1995). With the advent of Wikipedia, automatic techniques to extract knowledge from semi-structured text have successfully been applied by YAGO (Suchanek et al., 2007) and DBpedia (Auer et al., 2007).

Agichtein and Gravano (2000), Etzioni et al. (2004), and Carlson et al. (2010) employed semi-supervised methods using bootstrapping techniques together with initial seed relations in order to extract relations from unstructured text. Banko and Etzioni (2007) and Fader et al. (2011) refined them further using unsupervised approaches. Together, these approaches have successfully handled scalability and precision factors, when applied to unstructured text in web-scale corpora. Although syntactic and semantic parsers reach higher recalls and precisions (Christensen et al., 2010), their use in knowledge extraction has been relatively limited as they require more computing capabilities.

The identification of name mentions that are part of semantic propositions and their disambiguation is essential to link the resulting relations to external knowledge graphs. Examples of such an identification include Cattoni et al. (2012) that applied a named entity tagger and a coreference solver to systematically extract entities across a set of newspaper articles and LODifier (Augenstein et al., 2012) that combined a named entity disambiguator and a semantic parser to generate linked data. Prismatic (Fan et al., 2012) used a slot-grammar-based parser together with a named entity recognizer and a coreference solver to create a large-scale knowledge base.

Similarly to the Prismatic approach, we employ a combination of NLP tools to extract semantic units called frames. In addition, we extend these frames with a layer of semantic roles and we handle scalability through the Hadoop-based REFRACTIVE framework that we describe here.

3. Terminology

Following the terminology found in Fan et al. (2012), REFRACTIVE uses the definitions and extensions below:

Frames: We consider a frame as a representation of a semantic n-ary relation. Like Prismatic, we define its contents as a set of slot and value pairs. Frames in REFRACTIVE are instantiated by the syntactic and semantic representations of a sentence obtained from dependency parsing.

Slots: Slots represent binary relations. These are most commonly syntactic and semantic dependency relations.

Slot values: Prismatic defines a slot value as either the lemma form of a term from a sentence or a named entity type. We extend this definition by allowing values obtained from a coreference resolver, term yields, and semantic roles.

Frame projections: A frame projection is a subset of a frame. It is generated by specifying an ordered list of slots, such as SBJ-VERB-OBJ.

4. System Architecture

4.1. Overview

REFRACTIVE consists of a pipeline that takes Wikipedia articles as input and produces frame projections in the form of RDF triples and queryable Lucene indices. We first parse the Wikipedia articles using KOSHIK (Exner and Nugues, 2014), a MapReduce-based, open-source framework that includes an ensemble of natural language processing tools. We then extract the frames from the syntactic and semantic dependency representations of the article sentences. We finally generate frame projections or subsets using a predefined list of slots.

The instances of the frame projections can then be queried using an interface to a Lucene-based index. For each instance, we compute aggregate statistics based on frequency or conditional probability. As an alternative to Lucene indices, REFRACTIVE can also export data as RDF triples. For scalability, we run both frame extraction and projection as Map jobs in Hadoop.

Figure 1 shows an overview of the system architecture. In the following sections, we describe the modules that are part of the system in more detail.

4.2. Corpus Processing

We process and annotate the articles using KOSHIK[1]. KOSHIK combines tools for syntactic parsing (Bohnet, 2010), semantic parsing (Björkelund et al., 2009), named entity recognition, and coreference resolution (Raghunathan et al., 2010; Lee et al., 2011) within the Hadoop framework for distributed computing. This allows for the possibility to scale the NLP tasks with varying sizes of corpora by running atop of a cluster of commodity hardware. In addition, Hadoop comes with a rich set of exiting tools that KOSHIK can integrate and utilize.

KOSHIK annotates the documents using an extended version of the CoNLL 2008 format, where we added columns to support named entities and coreference chains. This enables the system to gather results with varying output formats into one single structure.

Using tools like Hive (Thusoo et al., 2009), the annotated corpus can then be queried using an interactive language. Hive offers an SQL-like language, called HiveQL, to express queries that are transformed into MapReduce jobs. Hive simplifies the analysis phase by offering a simplified querying language familiar to RDBMS analysts.

4.3. Frame Extraction

We extract frames from syntactic trees by projecting subtrees of fixed height. Slots are formed from each binary syntactic relation. We only consider frames having a noun or a verb as root. Following Prismatic, we limit the height of frames to a depth of two levels. This allows us to focus on the immediate participants in a frame. Furthermore, given that all dependency parsers make mistakes, this limitation decreases the probability that a participant has been incorrectly attached.

We attach a set of slot values to each slot. At a minimum, REFRACTIVE needs only the syntactic annotation of a document to extract frames. Optionally, the values of slots may be extended by annotating the document with semantic roles, coreference chains, and named entity tags. The lemma and yield of the term, restricted to the words inside the frame, are always attached together with a flag indicating if the term is a proper noun. Given a coreference chain, we resolve terms that constitute an anaphoric mention to their corresponding antecedent mention and attach this mention to the slot. When given, we also attach a named entity type as a slot value. Similarly to syntactic relations, semantic roles from sentences annotated with the semantic parser are transformed into slots.

Figures 2 and 3 show the syntactic and semantic outputs of the parser for the sentence:

> Wilhelm Conrad Röntgen won the first Nobel Prize in Physics in 1901 for his discovery of X-Rays,

while Figure 4 shows of how the syntactically annotated sentence is divided into frames. Table 1 shows the resulting frames extracted from the sentence.

4.4. Frame Projection

We create frame projections or subsets of frames by specifying an ordered list of slots, where each slot consists of a name and a value. The slot name is made of a dependency label, either syntactic like `SBJ`, or semantic like `WIN.01_A0`, and an optional modifier specifying its content. The modifier can either be: yield (`-Y`), type (`-T`), or coreference (`-C`).

- The yield modifier specifies that the value must be member of the yield, restricted to the words inside the frame and setting aside the determiners and prepositions (yield segment);

[1] Available at: `https://github.com/peterexner/KOSHIK/`

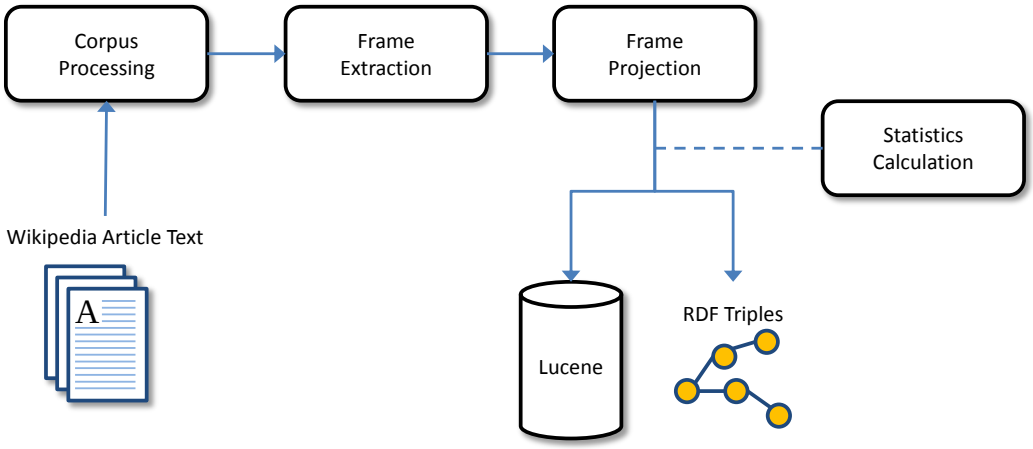

Figure 1: Overview of the REFRACTIVE knowledge extraction architecture.

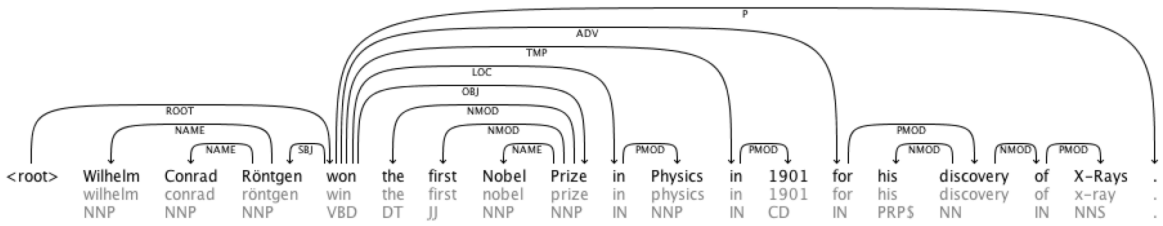

Figure 2: Dependency graph of the sentence *Wilhelm Conrad Röntgen won the first Nobel Prize in Physics in 1901 for his discovery of X-Rays*.

- the type indicates that the value must belong to a specific category, where the categories can be either: PERSON, LOCATION, ORGANIZATION, DATE, MONEY, NUMBER, ORDINAL, DURATION, or MISC.
- the coreference modifier specifies that the slot value must be a member of the coreference chain; the coreference chain is defined by the set of all its mentions.

For instance, the projection specification "SBJ-Y:Y, VERB, OBJ-Y:Y" describes frames with three slots: the yield segment of a subject slot, the lemma of a verb slot, and the yield segment of an object slot. The optional suffix `:Y` indicates that both the subject and the object slots are constrained to contain only proper nouns. REFRACTIVE instantiates this projected frame with slot values from frames that contain a matching set of slots.

4.5. Frame Statistics

REFRACTIVE supports the calculation of two types of frame statistics: frequency and conditional probability. All the statistics are calculated from the resulting frame instances generated by the frame projections.

The frame frequency module calculates the number of times an occurrence of a specific set of slot values has been found in the corpus. For instance, it might be of interest to find out how many times a "SBJ, VERB, OBJ" projection with the values "*Yankees*, *win*, *cup*" occurs in a text.

The conditional probability is calculated by specifying two frame projections: a target projection and a conditional projection. For instance, given a conditional projection, "SBJ, VERB", and a target projection, "SBJ, VERB, OBJ", REFRACTIVE calculates the conditional probability that a certain object value occurs given certain values of subject and verb. Table 2 shows an example of a conditional projection and how conditional probabilities are useful in finding the countries that the United States most probably annexed. Similarly, conditional probabilities may be used to generate probable answers to other questions.

Subject	Verb	Object	Probability
United States	annex	Texas	0.3
United States	annex	Puerto Rico	0.1
United States	annex	Hawaii	0.07

Table 2: An example of a "SBJ-Y:Y, VERB, OBJ-Y:Y" projection with the values SBJ:*United States* and VERB:*annex*. Conditional probability statistics based on the conditional frame projection, "SBJ-Y:Y, VERB", have been calculated for each frame instance.

4.6. Export and Querying

REFRACTIVE supports the export of frame projections to Lucene indices and RDF triples. Each frame projection instance is stored as a document in Lucene with fields set by the slot values from the projection. This enables REFRACTIVE to generate answers by querying a Lucene index with a set of known slot values. For instance, given a Lucene index containing instances of "SBJ-Y:Y, VERB, OBJ-Y:Y" projections, it is possible to generate a list of Nobel Prize winners by querying "VERB:*win*, OBJ:*Nobel Prize*".

	Wilhelm	Conrad	Röntgen	won	the	first	Nobel	Prize	in	Physics	in	1901	for	his	discovery	of	X-Rays	.
win.01	A0				A1				AM-LOC		AM-TMP		AM-CAU					
discovery.01														A0			A1	

Figure 3: Semantic propositions extracted from the sentence *Wilhelm Conrad Röntgen won the first Nobel Prize in Physics in 1901 for his discovery of X-Rays*.

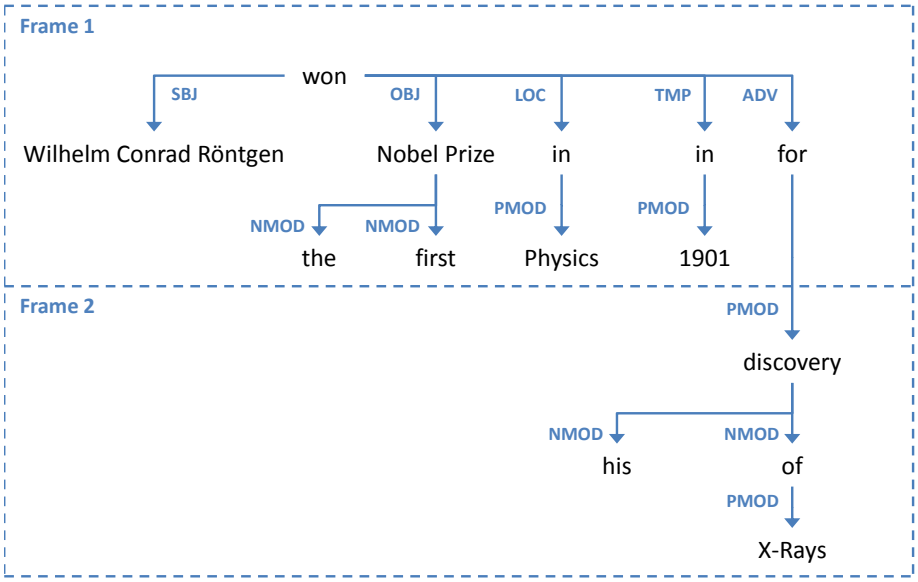

Figure 4: Frames extracted from the sentence *Wilhelm Conrad Röntgen won the first Nobel Prize in Physics in 1901 for his discovery of X-Rays*.

5. Experimental Results

We processed over 4 million articles from the English edition of Wikipedia and we extracted more than 260 million frames. On average, this amounts to 4.3 frames per sentence. To carry out the syntactic and semantic analyses, KOSHIK uses a high precision, graph-based dependency parser (Bohnet, 2010). This parser achieved an F1-score of 90.33 on the CoNLL 2009 shared task corpus at the expense of a relatively slow speed. Using a 60-core cluster, corpus processing and frame extraction took 463 hours.

We counted the number of predicates covered by our extracted frames and we found that the frames include 95.5% of the extracted predicates. Similarly, we compared the coverage of named entity tags by the extracted frames and found that 94.9% of the named entities are included in some frames. Table 3 summarizes the statistics we extracted from Wikipedia.

Property	**Value**
Corpus size	7.6 GB
Sentences	61,265,766
Frames extracted	263,586,849
Frames per sentence	4.3
Predicate coverage	95.5%
Named entity coverage	94.9%

Table 3: Statistics extracted from Wikipedia.

6. Conclusions and Future Work

In this paper, we described REFRACTIVE – a tool for extracting knowledge from unstructured text. We represent the extracted knowledge as semantic n-ary relations, described by a frame with a set of slot and slot values. We implemented a scalable framework for processing articles using tools that can extract, project, and query frames. By running REFRACTIVE on the English edition of Wikipedia, we have extracted over 260 million frames, covering more than 94% of the named entities and predicates present in the articles.

In the future, we plan to use REFRACTIVE for distant supervision of semantic role labeling. While many languages have corpora for training syntactic parsers, fewer have corpora of sufficient size annotated with semantic roles. We believe that by generating frames for parallel corpora in two different languages and matching entities in parallel frames, semantic roles can be transferred from one language to another. Figure 5 shows an example of such a situation.

The source code for REFRACTIVE is available for download at `https://github.com/peterexner/REFRACTIVE/`.

7. Acknowledgments

This research was supported by Vetenskapsrådet, the Swedish research council, under grant 621-2010-4800, and the *Det digitaliserade samhället* and eSSENCE programs.

Frame 1					
Slot	**Proper noun**	**Lemma**	**Coreference**	**Yield segment**	**Named entity type**
SBJ	Y	röntgen		Wilhelm Conrad Röntgen	PERSON
VERB	N	win		Wilhelm ... won .. discovery	
NMOD	N	the			
NMOD	N	first		first	ORDINAL
OBJ	Y	prize		first Nobel Prize	
LOC	N	in		Physics	
PMOD	Y	physics		Physics	
TMP	N	in		1901	
PMOD	N	1901		1901	DATE
ADV	N	for		discovery	
PMOD	N	Frame 2		discovery	
WIN.01_A0	Y	röntgen		Wilhelm Conrad Röntgen	PERSON
WIN.01_A1	Y	prize		first Nobel Prize	
WIN.01_AM-LOC	N	in		Physics	
WIN.01_AM-TMP	N	in		1901	
WIN.01_AM-CAU	N	for		discovery	
Frame 2					
Slot	**Proper noun**	**Lemma**	**Coreference**	**Yield segment**	**Named entity type**
NMOD	N	his	Wilhelm Conrad Röntgen		
NOUN	N	discovery		his discovery X-Rays	
NMOD	N	of		X-Rays	
PMOD	N	x-ray		X-Rays	
DISCOVERY.01_A0	N	his	Wilhelm Conrad Röntgen		
DISCOVERY.01_A1	N	of		X-Rays	

Table 1: The two frames extracted from the sentence: *Wilhelm Conrad Röntgen won the first Nobel Prize in Physics in 1901 for his discovery of X-Rays*.

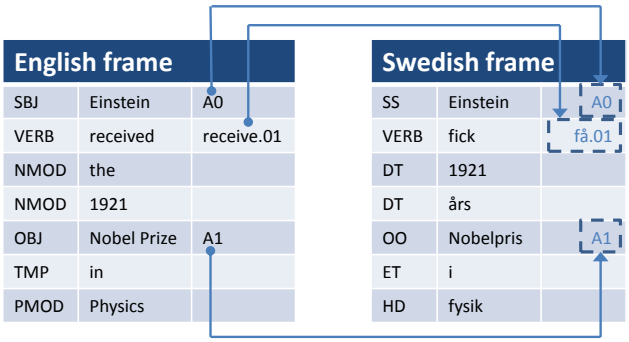

Figure 5: Using frames to create a training corpus for semantic role labeling.

8. References

Agichtein, E. and Gravano, L. (2000). Snowball: extracting relations from large plain-text collections. In *Proceedings of DL '00*, pages 85–94, New York. ACM.

Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., and Ives, Z. (2007). DBpedia: A nucleus for a web of open data. In *The Semantic Web*, volume 4825 of *LNCS*, pages 722–735. Springer Berlin.

Augenstein, I., Padó, S., and Rudolph, S. (2012). LODifier: Generating linked data from unstructured text. In *The Semantic Web: Research and Applications*, volume 7295 of *LNCS*, pages 210–224. Springer Berlin.

Banko, M. and Etzioni, O. (2007). Strategies for lifelong knowledge extraction from the web. In *Proceedings of K-CAP '07*, pages 95–102, New York. ACM.

Björkelund, A., Hafdell, L., and Nugues, P. (2009). Multilingual semantic role labeling. In *Proceedings of The Thirteenth Conference on Computational Natural Language Learning (CoNLL-2009)*, pages 43–48, Boulder, June 4-5.

Björkelund, A., Bohnet, B., Hafdell, L., and Nugues, P. (2010). A high-performance syntactic and semantic dependency parser. In *Coling 2010: Demonstration Volume*, pages 33–36, Beijing, August 23-27. Coling 2010 Organizing Committee.

Bohnet, B. (2010). Top accuracy and fast dependency parsing is not a contradiction. In Huang, C.-R. and Jurafsky, D., editors, *COLING*, pages 89–97. Tsinghua University Press.

Carlson, A., Betteridge, J., Kisiel, B., Settles, B., Hruschka, E. R., and Mitchell, T. M. (2010). Toward an architecture for never-ending language learning. In *Proceedings of AAAI-10*, pages 1306–1313.

Cattoni, R., Corcoglioniti, F., Girardi, C., Magnini, B., Serafini, L., and Zanoli, R. (2012). The KnowledgeStore: an entity-based storage system. In *Proceedings of LREC 2012*, Istanbul, may.

Christensen, J., Mausam, Soderland, S., and Etzioni, O. (2010). Semantic role labeling for open information extraction. In *Proceedings of the NAACL HLT 2010 First International Workshop on Formalisms and Methodology for Learning by Reading*, FAM-LbR '10, pages 52–60.

Etzioni, O., Cafarella, M., Downey, D., Kok, S., Popescu, A.-M., Shaked, T., Soderland, S., Weld, D. S., and Yates, A. (2004). Web-scale information extraction in knowitall. In *Proceedings of WWW '04*, pages 100–110, New York. ACM.

Exner, P. and Nugues, P. (2014). KOSHIK: A large-scale distributed computing framework for nlp. In *Proceedings of ICPRAM 2014 – The 3rd International Conference on Pattern Recognition Applications and Methods*, pages 464–470, Angers, March 6-8.

Fader, A., Soderland, S., and Etzioni, O. (2011). Identifying relations for open information extraction. In *Proc. of EMNLP '11*, pages 1535–1545.

Fan, J., Ferrucci, D., Gondek, D., and Kalyanpur, A. (2010). Prismatic: Inducing knowledge from a large scale lexicalized relation resource. In *Proceedings of the NAACL HLT 2010 First International Workshop on Formalisms and Methodology for Learning by Reading*, pages 122–127.

Fan, J., Kalyanpur, A., Gondek, D. C., and Ferrucci, D. (2012). Automatic knowledge extraction from documents. *IBM Journal of Research and Development*, 56(3.4):5:1–5:10.

Ferrucci, D. A. (2012). Introduction to "This is Watson". *IBM Journal of Research and Development*, 56(3.4):1:1 –1:15, may-june.

Lee, H., Peirsman, Y., Chang, A., Chambers, N., Surdeanu, M., and Jurafsky, D. (2011). Stanford's multi-pass sieve coreference resolution system at the CoNLL-2011 shared task. In *Proceedings of the CoNLL-2011 Shared Task*, pages 28–34, Boulder.

Lenat, D. B. (1995). Cyc: A large-scale investment in knowledge infrastructure. *Communications of the ACM*, 38(11):33–38.

Mausam, Schmitz, M., Soderland, S., Bart, R., and Etzioni, O. (2012). Open language learning for information extraction. In *Proceedings of the 2012 Joint Conference on EMNLP and CoNLL*, pages 523–534, Jeju Island, July.

Miller, G. and Fellbaum, C. (1998). *Wordnet: An electronic lexical database*. MIT Press Cambridge.

Palmer, M., Gildea, D., and Kingsbury, P. (2005). The Proposition Bank: an annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71–105.

Raghunathan, K., Lee, H., Rangarajan, S., Chambers, N., Surdeanu, M., Jurafsky, D., and Manning, C. (2010). A multi-pass sieve for coreference resolution. In *Proc. of EMNLP-2010*, pages 492–501, Boston.

Ruppenhofer, J., Ellsworth, M., Petruck, M. R. L., Johnson, C. R., and Scheffczyk, J. (2010). Framenet ii: Extended theory and practice. `http://framenet.icsi.berkeley.edu/`, September.

Singhal, A. (2012). Introducing the knowledge graph: things, not strings. Official Google Blog, May.

Suchanek, F. M., Kasneci, G., and Weikum, G. (2007). Yago: A Core of Semantic Knowledge. In *16th international World Wide Web conference (WWW 2007)*, New York, NY, USA. ACM Press.

Thusoo, A., Sarma, J. S., Jain, N., Shao, Z., Chakka, P., Anthony, S., Liu, H., Wyckoff, P., and Murthy, R. (2009). Hive: a warehousing solution over a map-reduce framework. *Proceedings of the VLDB Endowment*, 2(2):1626–1629.