An Overview of Knowledge Extraction Projects in the NLP group at Lund University

Pierre Nugues
Department of Computer Science
Lund University, Lund, Sweden
Pierre.Nugues@ce.lth.se

Abstract

In this paper, I describe systems and prototypes we created in the natural language processing group at Lund to extract structured knowledge from text. Starting from syntactic and semantic parsing components, we developed applications that can handle large corpora, typically complete Wikipedia versions consisting of millions of documents and process text to identify entities and the relations between them. I describe the overall goals of our projects, the data structure we designed to handle the documents, as well as three applications to extract knowledge from text.

1 Question-Answering Systems for Swedish and Other Languages

The multiple digitization initiatives make larger and larger quantities of text everyday more accessible. Within one or two decades, we can imagine that most of what has been written and made public in a printed form will be available in a machine-readable format to anyone with an internet connection. Bill Gates prediction of *information at your fingertips*, made in November 1990, will have come to a reality.

Large parts of the human knowledge are crystalized in text and given its accessibility in a digital form, machines can automatically extract it and process it. The recent success of question-answering systems like IBM Watson (Ferrucci, 2012) that answer questions better than human beings in quiz contests is a proof of it. Text is in fact the raw material of question-answering systems and Watson that builds on the whole Wikipedia collection and documents retrieved from the web, is a dramatic achievement of automatic knowledge extraction.

Most efforts in the development of knowledge extraction systems focus on English. See the evaluations of the Text Retrieval Conferences (TREC)[1], for instance. Such a focus may eventually overlook many sources in other languages and impoverish human knowledge in general. From the beginning, starting from Swedish, our group tried to develop multilingual systems.

In its simplest form, the structure of a question–answering system consists of three components (Fig 1, simplified from Watson):

1. A question processing module that analyzes the question, identifies the entities, and predicts the answer type, a person, location, etc.;

2. A passage retrieval module that builds on knowledge sources, large text repositories, and indexes them. Given a question, this module produces a list of passages that hopefully contain the answer;

3. Finally, an answer extraction module that extracts answers from the passages and ranks them.

We conducted pilot implementations of a question-answering system for Swedish to see how this architecture could generalize. Using a corpus of questions inspired by the Swedish television quiz show Kvitt eller Dubbelt – Tiotusenkronorsfrågan (Thorsvad and Thorsvad 2005; Kvitt eller Dubbelt 2013). (Thorsvad and Thorsvad, 2005; Kvi, 2013), we evaluated the coverage of the Swedish version of Wikipedia as knowledge source (Pyykkö et al., 2014). We split Wikipedia into paragraphs; we indexed them with the Lucene tool; and given a question, we ranked the paragraphs with the $TF.IDF$ measure.

[1] http://trec.nist.gov

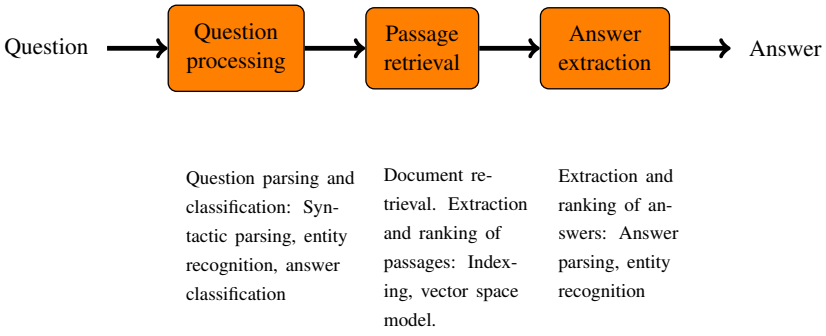

Figure 1: Overall architecture of a question–answering system, simplified from Ferrucci (2012)

We found that about 90% of the answers are in one or more passages of Wikipedia and about 70% in the 200 first passages returned by a $TF.IDF$ ranking. In a second experiment, using the parts of speech of the words in the passages, an answer type predictor, and the category of the proper nouns, we could extract answers with a median rank of the correct answer of 21; we could improve this rank to 10 with a reranker (Grundström and Nugues, 2014). These results showed that the Swedish Wikipedia, although much smaller than the English counterpart, was a viable and valuable knowledge source for question answering systems. This also hinted at a probable replicability of the results to other languages.

2 Propositions Databases

In addition to passages, we can extract structured propositions, consisting of predicates and arguments, from text. For example, from the Wikipedia excerpt:

> Shakespeare was born and brought up in Stratford-upon-Avon, Warwickshire,

we can derive the two predicate–argument structures:

```
born(Shakespeare, Stratford-upon-Avon\, Warwickshire)
brought_up(Shakespeare, Stratford-upon-Avon\, Warwickshire).
```

that convert this piece of text into database facts. Repositories of such facts extracted from large corpora usually improve the performance of question–answering systems – 2.4% in the case of Watson –.

Exner and Nugues (2012) applied a semantic role labeler (Johansson and Nugues, 2008; Björkelund et al., 2010) to the whole English Wikipedia to automatically derive a set of predicate–argument structures. This eventually resulted in more than 260 million propositions (Exner and Nugues, 2014). Figure 2 shows the web interface to the Athena database, where a user can enter a predicate and up to four arguments. In Fig. 2, the user has entered the predicate *kill* and the direct object *bacteria* corresponding to the question *What kills bacteria?* and the system retrieves all the propositions matching the predicate:

```
kill(X, bacteria)
```

where `X` is a variable, (A0 in Fig. 2), yielding `X = antibiotics`, `X = heat`, `X = systems`, `X = acids`, etc.

3 Multilingual Propositions Databases

While semantic role labelers are generic tools to extract predicate–argument structures, they are language-dependent and require to be trained on large manually-annotated resources. There are no such large annotated corpora for Swedish and many other languages, including French as of the date this paper is written. We developed a tool to project propositions across languages. Fillmore (1976) gives an argument on the crosslingual nature of predicate–argument structures (frames):

> A particularly important notion [...] that goes by such names as “frames”, [...]. Briefly, the idea is that people have in memory an inventory of schemata for structuring, classifying, and

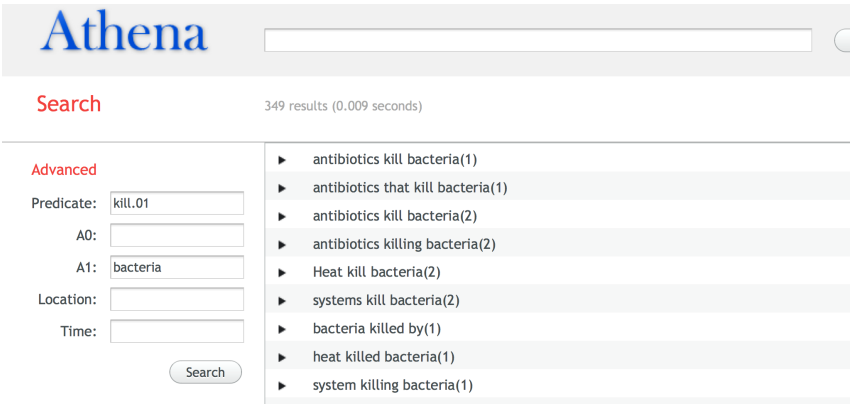

Figure 2: A screenshot of the Athena system (Exner and Nugues, 2012)

> interpreting experiences, [...] The concept of frame does not depend on language, but as applied to language processing the notion figures in the following way. Particular words or speech formulas, or particular grammatical choices, are associated in memory with particular frames

To extract and abstract such frames across languages, we applied the following ideas:

1. Ground the frames (schematas) in reality through their actors (arguments), independently from the language;

2. Use real world entities, such as Aristotle or *the Organon*, to identify the actors more easily;

3. Find the predicates or verbal nodes connecting these actors.

As an example, the sentence *Aristotle wrote the Organon* has 32 occurrences in Google books[2], while the equivalent French sentence *Aristote a écrit l'Organon* has 3. Recognizing the two named entities, Aristotle and Organon, in propositions, if frequent enough, will probably involve the same relation: write/écrire (Fig. 3) with its two core arguments in Framenet: *author* and *text* (Ruppenhofer et al., 2005). It will be then possible to derive an annotated corpus of relations in French.

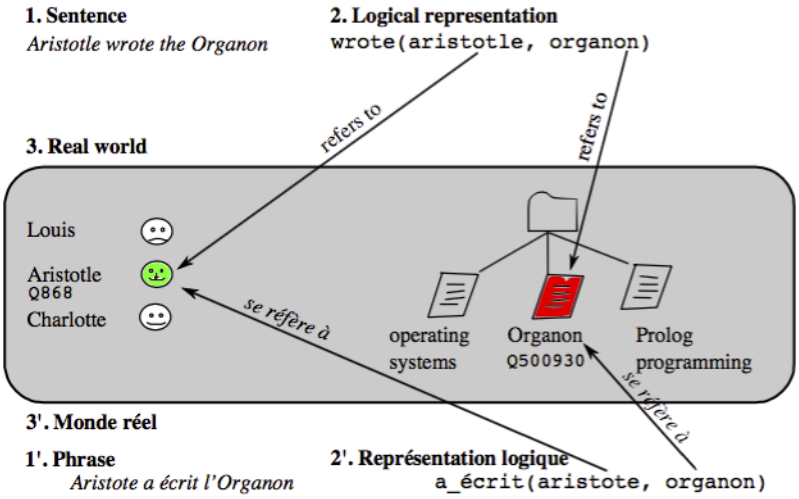

Figure 3: Crosslingual projection of predicate–argument structures

[2]Retrieved on April 7, 2016

Exner et al. (2015) used the Swedish and English versions of Wikipedia to collect a large set of parallel propositions. They identified named entities in these two versions using a unique identifier, the Wikidata Q-number, that enabled them to pair the predicate–argument structures across the languages.

In Wikipedia, a same entity or concept can have from one to more than 200 different language versions. Wikidata is a graph database that connects all these entities and concepts across the versions. It has the form of a centralized repository that stores links to all the versions with a unique number: Q868 in the case of Aristotle and Q500930 for the Organon (Fig. 4).

Figure 4: Left: The first language versions of Aristotle in Wikidata. The languages in the figure appear in alphabetic order out of 171. Right: The first language versions of the Organon out of 32 entries

In addition to listing entities, Wikidata uses a set of about 2,000 properties[3] to describe them. Aristotle, for example, is an instance of a human (Fig. 5), where *instance of* property, P31, enables the editors to define an ontology.

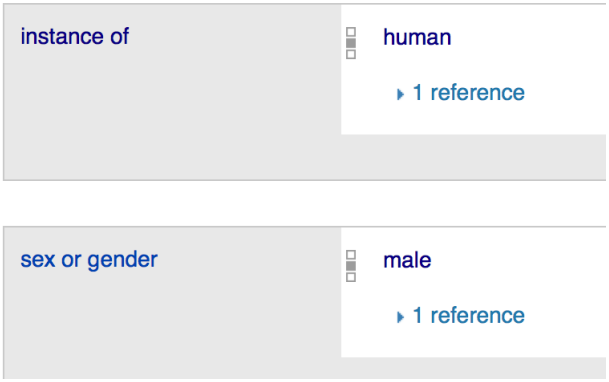

Figure 5: Membership of Aristotle to ontology classes using the *instance of* property

Using the resulting propositions in Swedish, Exner et al. (2015) could train a semantic role labeler. While not on a par with semantic role labelers trained on large hand-annotated corpora, it obtains promising results and in fact can identify the arguments of the Swedish sentence:

Aristoteles har skrivit Organon,
'Aristotle wrote the Organon'

although no such a sentence exists in the Web (Fig. 6).

[3] 1,863 properties as of October 12, 2015

	Aristoteles	har	skrivit	Organon	.
skriva.01	A0			A1	

Parsing sentence required 6ms.

Figure 6: Semantic parsing of *Aristoteles har skrivit Organon*. The arguments are given using the Propbank nomenclature (Palmer et al., 2005), where A0 is the *writer* and A1 is the *thing written*

4 Scaling to Large Corpora

While small corpora can be handled in the form of files, the size of Wikipedia requires a different data structure. We created a document model to store large collections of text. We designed it so that we could store annotations such as token, sentence, paragraph, etc., keep the wiki markup, as well as the subsequent linguistic annotations. Figure 7 shows the structure of this model, Docforia, that consists of multiple layers as well as the conversion process from Wikimedia dumps (Klang and Nugues, 2016). The ideas behind Docforia are similar to those of the UIMA project (Ferrucci and Lally, 2004), but the focus is on simplicity and ease of integration.

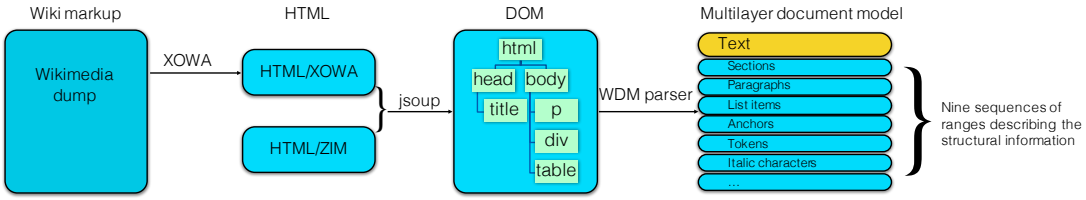

Figure 7: Docforia: A multilayer document model. After Klang and Nugues (2016)

The source code of Docforia as well as its documentation are available from `https://github.com/marcusklang/docforia`.

5 Extraction of Career Timelines

Finally, the retrieval of career timeline is an example of application of knowledge extraction from Wikipedia. Using the Swedish version and the Wikidata ontology, Dib et al. (2015) could extract the careers of people. They restricted the pages to people and analyzed the first paragraph with a dependency parser to find grammatical links between the person described by the page and professions defined as subclasses the Wikidata occupation property. Figure 8 shows an example of sentence parsing to link Göran Persson, a Swedish Prime minister, to *politiker* 'politician' and *statsminister* 'Prime Minister'.

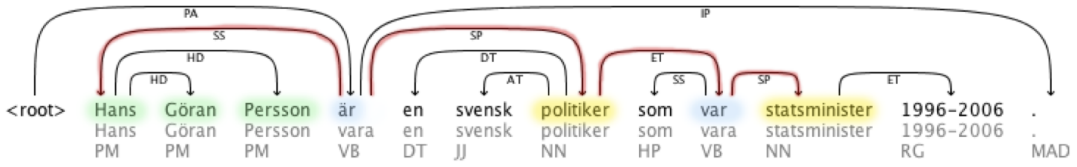

Figure 8: Linking a person to occupations through a dependency graph. After Dib et al. (2015)

Figure 9 shows a screenshot of the application interface.

Career profile of "Göran Persson"

Göran Persson (5)

- Skolminister (1989-1991) Based on the verb "vara"

  Han var skolminister 1989-91, finansminister 1994-96, riksdagsledamot 1979-84 och 1991-2007 samt ledamot av socialdemokratiska partistyrelsen och partiordförande 1996-2007.

- Finansminister (1994-1996) Based on the verb "vara"

  Han var skolminister 1989-91, finansminister 1994-96, riksdagsledamot 1979-84 och 1991-2007 samt ledamot av socialdemokratiska partistyrelsen och partiordförande 1996-2007.

- Statsminister (1996-2006) Based on the verb "vara"
- Svensk politiker Based on the verb "vara"
- Talesperson Based on the verb "vara"

Figure 9: The career timeline of Göran Persson

Acknowledgements

This research was supported by Vetenskapsrådet, the Swedish research council, under the *Det digitaliserade samhället* program.

References

Anders Björkelund, Bernd Bohnet, Love Hafdell, and Pierre Nugues. 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.

Firas Dib, Simon Lindberg, and Pierre Nugues. 2015. Extraction of career profiles from Wikipedia. In *BD2015, Proceedings of the First Conference on Biographical Data in a Digital World 2015*, pages 33–38, Amsterdam, April. CEUR Workshop Proceedings.

Peter Exner and Pierre Nugues. 2012. Constructing large proposition databases. In *Proceedings of the eighth international conference on Language Resources and Evaluation (LREC 2012)*, pages 3836–3840, Istanbul, May 23–25.

Peter Exner and Pierre Nugues. 2014. REFRACTIVE: An open source tool to extract knowledge from syntactic and semantic relations. In *Proceedings of LREC 2014, The 9th edition of the Language Resources and Evaluation Conference*, pages 2584–2589, Reykjavik, May 27-29.

Peter Exner, Marcus Klang, and Pierre Nugues. 2015. A distant supervision approach to semantic role labeling. In *Proceedings of the Fourth Joint Conference on Lexical and Computational Semantics*, pages 239–248, Denver, Colorado, June. Association for Computational Linguistics.

David Ferrucci and Adam Lally. 2004. Uima: An architectural approach to unstructured information processing in the corporate research environment. *Nat. Lang. Eng.*, 10(3-4):327–348, September.

David Angelo Ferrucci. 2012. Introduction to “This is Watson”. *IBM Journal of Research and Development*, 56(3.4):1:1 –1:15, May-June.

Charles J. Fillmore. 1976. Frame semantics and the nature of language. *Annals of the New York Academy of Sciences: Conference on the Origin and Development of Language and Speech*, 280:20–32.

Jakob Grundström and Pierre Nugues. 2014. Using syntactic features in answer reranking. In *Proceedings of the AAAI 2014 Workshop on Cognitive Computing for Augmented Human Intelligence*, pages 13–19, Québec, July 27.

Richard Johansson and Pierre Nugues. 2008. Dependency-based semantic role labeling of PropBank. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing (EMNLP 2008)*, pages 69–78, Honolulu, October 25-27.

Marcus Klang and Pierre Nugues. 2016. Wikiparq: A tabulated Wikipedia resource using the Parquet format. In *Proceedings of 10th edition of the Language Resources and Evaluation Conference*, Portorož, May.

2013. Kvitt eller dubbelt – tiotusenkronorsfrågan. `http://en.wikipedia.org/wiki/Kvitt_eller_dubbelt`.

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

Juri Pyykkö, Rebecka Weegar, and Pierre Nugues. 2014. Passage retrieval in a question answering system. In *Proceedings of the The Fifth Swedish Language Technology Conference (SLTC 2014)*, Uppsala, November 13-14.

Josef Ruppenhofer, Michael Ellsworth, Miriam R. L. Petruck, and Christopher R. Johnson. 2005. Framenet: Theory and practice. `http://framenet.icsi.berkeley.edu/book/book.html`. Cited 28 October 2005.

Karin Thorsvad and Hasse Thorsvad. 2005. Kvitt eller dubbelt.