Identification of Entities in Swedish

Salomonsson, Andreas; Marinov, Svetoslav; Nugues, Pierre

Published in:
SLTC 2012

2012

Identification of Entities in Swedish

Andreas Salomonsson* **Svetoslav Marinov*** **Pierre Nugues†**

*Findwise AB, Drottninggatan 5
SE-411 14 Gothenburg, Sweden
{andreas.salomonsson,svetoslav.marinov}@findwise.com
†Department of Computer Science, Lund University
S-221 00 Lund, Sweden
pierre.nugues@cs.lth.se

1. Introduction

A crucial aspect of search applications is the possibility to identify named entities in free-form text and provide functionality for entity-based, complex queries towards the indexed data. By enriching each entity with semantically relevant information acquired from outside the text, one can create the foundation for an advanced search application. Thus, given a document about Denmark, where neither of the words *Copenhagen*, *country*, nor *capital* are mentioned, it should be possible to retrieve the document by querying for *capital Copenhagen* or *European country*.

In this paper, we report how we have tackled this problem. We will, however, concentrate only on the two tasks which are central to the solution, namely named entity recognition (NER) and enrichment of the discovered entities by relying on linked data from knowledge bases such as YAGO2 and DBpedia. We remain agnostic to all other details of the search application, which can be implemented in a relatively straight-forward way by using, e.g. Apache Solr[1].

The work deals only with Swedish and is restricted to two domains: news articles[2] and medical texts[3]. As a byproduct, our method achieves state-of-the-art results for Swedish NER and to our knowledge there are no previously published works on employing linked data for Swedish for the two domains at hand.

2. Named Entity Recognition for Swedish

Named entity recognition is an already well-established research area with a number of conferences and evaluations dedicated to the task (e.g. CoNLL 2003 (Tjong et al., 2003)). While many systems have been created for English and other languages, much fewer works have been reported on Swedish.

Dalianis and Åström (2001) and Johannessen et al. (2005) are examples of NER systems for Swedish. Dalianis and Åström (2001) employ rules, lexicons, and machine–learning techniques in order to recognize four types of entities: person, location, organization, and time. It achieves an overall F-score of 0.49 on 100 manually tagged news texts.

The Swedish system described in Johannessen et al. (2005) is rule-based, but relies on shallow parsing and employs gazetteers as well. It can recognize six types of entities: person, organization, location, work, event, and other. It reports an F-score of 0.92 measured on test corpus of 1800 words. On a test set of 40 000 words without gazetteers, the recall of the system drops from 0.91 to 0.53. No information of precision is given by the authors.

3. System Description

3.1 Named Entity Recognition

We have tackled the NER problem by relying entirely on machine–learning techniques. We use a linear classifier, LIBLINEAR (Fan et al., 2008), and train and test the system on the Stockholm-Umeå Corpus 2.0 (SUC) (Ejerhed et al., 2006).

In addition to the POS tags of the tokens, SUC contains information about nine different categories of named entities: person, place, institution, work, animal, product, myth, event, and other. Following the standards in CoNLL 2003, we chose to identify four categories: person, organization, location, and miscellaneous, thus merging the product, myth, event, animal, work and other classes in a miscellaneous category and mapping institution to organization, and place to location.

The most important features of the classifier are: the POS tags of the surrounding and the current token, the word tokens themselves, and the previous two named entity tags. Other features include Booleans to describe the initial capitalization, if the word contains periods, and contains digits. As advocated by Ratinov and Roth (2009), we employ the BILOU (Begin Inside Last Outside Unique) annotation for the entities.

3.2 Linked Data

Linked Data is a part of the semantic web initiative. In its core, it consists of concepts interlinked with each other by RDF triples. This allows us to augment the discovered entities with additional information related to them. We have used the semantic network YAGO2 (Hoffart et al., 2011) and the DBpedia knowledge base. Each named entity we extract from the NER module is mapped to an identifier in YAGO2 if the entity exists in the semantic network. We can then use information about the entity and its relations to other entities. YAGO2 is stored and queried using Sesame RDF repository from openRDF.org, and one can use SPARQL as the query language. One of the most important predicates in YAGO2 for our work is the `isCalled` predicate. Given an entity E we can

[1] http://lucene/apache.org/solr/
[2] Articles crawled from dn.se
[3] Articles from 1177.se

Category	Exist	Found	Correct	Precision (%)	Recall (%)	F_1 (%)
Persons	15128	17198	13626	78.17	90.47	83.87
Organizations	6332	4540	3089	68.33	47.49	56.03
Locations	8773	8974	6926	76.97	78.32	77.64
Miscellaneous	3956	2051	1249	64.73	29.60	40.62
Total	34189	32763	24890	75.77	72.35	74.02
Unlabelled	34189	32922	30655	93.11	89.66	91.36

Table 1: NER evaluation

use `SELECT ?id WHERE {?id isCalled "E"}?` to get one or several unique identifiers. If there is only one, we map the entity with its identifier. When multiple identifiers are found, we use a very simple method for disambiguation: We take the identifier with most information related to it.

While YAGO2 was used primarily with the news articles, DBPedia was employed for the medical texts. First each medical term in the SweMESH[4] taxonomy was mapped to an unique identifier from DBPedia. We then searched in the document only for those medical terms which are in SweMESH and use the unique identifier to extract more information about them.

4. Architecture Overview

The core of the system which identifies and augments named entities is implemented as a pipeline. The text of each document passes through the following stages: tokenization, sentence detection, POS tagging, NER, extraction of semantic information. At the end of the pipeline, the augmented document is sent for indexing.

The tokenizer was implemented by modifying the Lucene tokenizer used by Apache Solr. It uses regular expressions to define what a token is. The sentence detector is rule-based and uses information about the types of the tokens, and the tokens themselves as identified in the previous step. The POS tagging stage employs HunPos (Halácsy et al., 2007) which has been trained on SUC. The NER stage is the system which was described in Section 3.1 above. Finally, all named entities are enriched with semantic information if such is available from the knowledge bases.

Given the sentence *Danmark ligger i Nordeuropa.*, the two locations, *Danmark* and *Nordeuropa*, are identified. We then proceed by mapping the two entities to their corresponding identifiers in the semantic network. The YAGO2 identifier for *Danmark* is `http://www.mpii.de/yago/resource/Denmark` and we use it to extract information, e.g. Denmark is a European country; its capital is Copenhagen, with which we then augment the METAdata of the document.

5. Results and Discussion

We have evaluated the performance of the NER system with a 10-fold cross-validation on SUC. We used L2-regularized L2-loss support vector classification (primal) as the solver type in LIBLINEAR and all other parameters were set to their default values. While our system achieves very good results for Swedish (see Table 1), it is difficult to compare it to previous systems due to the differences in test data and number of categories. Yet, we see more room for improvement by adding gazetteers and using word clusters as features. In addition, we have noticed inconsistencies in the annotation of entities in SUC. Titles are sometimes part of the entities and sometimes not.

Finally, by enriching entities with related information allows us to retrieve more documents, cluster them in a better way or populate ontologies by using such data. As the extracted information resides in a META field of an indexed document and the field often gets a higher score, the document will get an overall higher rank in the result list.

[4] `http://mesh.kib.ki.se/swemesh/swemesh_se.cfm`

6. References

Hercules Dalianis and Erik Åström. 2001. SweNam–A Swedish named entity recognizer. Technical report, KTH.

Eva Ejerhed, Gunnel Källgren, and Benny Brodda. 2006. Stockholm-umeå corpus version 2.0.

Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. 2008. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874.

Péter Halácsy, András Kornai, and Csaba Oravecz. 2007. HunPos: an open source trigram tagger. In *Proceedings of the ACL 2007 Demo and Poster Sessions*.

Johannes Hoffart, Fabian M. Suchanek, Klaus Berberich, Edwin Lewis-Kelham, Gerard de Melo, and Gerhard Weikum. 2011. Yago2: Exploring and querying world knowledge in time, space, context, and many languages. In *Proceedings of the 20th International Conference Companion on World Wide Web (WWW 2011)*. ACM.

Janne Bondi Johannessen, Kristin Hagen, Åsne Haaland, Andra Björk Jónsdottir, Anders Nøklestad, Dimitris Kokkinakis, Paul Meurer, Eckhard Bick, and Dorte Haltrup. 2005. Named Entity Recognition for the Mainland Scandinavian Languages. *Literary and Linguistic Computing*, 20(1).

Lev Ratinov and Dan Roth. 2009. Design challenges and misconceptions in named entity recognition. In *CoNLL '09: Proceedings of the Thirteenth Conference on Computational Natural Language Learning*. ACL.

Erik F. Tjong, Kim Sang, and Fien De Meulder. 2003. Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. In *Proceedings of CoNLL-2003*.