The Saara Framework
An Anaphora Resolution System for Czech

Vašek Němčík

NLP Laboratory, Faculty of Informatics
Masaryk University Brno, Czech Republic
xnemcik@fi.muni.cz

Abstract. Determining reference and referential links in discourse is one of the biggest and most important challenges in natural language understanding. In particular, computing coreference classes over the set of referring expressions in text is crucial for its further syntactic and semantic processing. We present a system for automatic anaphora resolution that can be used on arbitrary texts in Czech. The article describes the individual phases of processing the input text and mentions selected issues that need to be addressed by the system.

Key words: anafora; anafora resolution; Czech language

1 Introduction

In this work, we present Saara (System for Automatic Anaphora Resolution and Analysis), a framework for anaphora resolution (AR) which is modular in many ways. Modularity in the context of AR has many obvious advantages. It allows defining various AR algorithms and using them for different purposes. Given a corpus annotated for coreference is available, it is possible to evaluate them, compare their strong and weak points and based on this knowledge, define and test more sophisticated ones. It is also possible to experiment with algorithms across genres or even languages.

In this paper, we mainly focus on utilizing the Saara framework as a part of a Natural Language Processing (NLP) system dealing with unrestricted Czech text on input. This mainly involves suitably combining it with pre-processing tools that perform the necessary linguistic analysis of the input text. These yield information required by AR algorithms to model phenomena relevant for anaphoric relations.

To our knowledge, there is currently no other AR system for Czech that can be straightforwardly used in an application setting to deal with texts that haven't been pre-processed manually. The only other AR system applicable to Czech data was presented by Linh [1], and to my knowledge, it can be used only with data manually annotated according to the three-layer formalism used within the Prague Dependency Treebank [2,3]. Our work aims at reaching a system that can used with arbitrary, unedited plain text. This addresses a crucial bottleneck in the practical applicability of NLP systems for Czech.

Petr Sojka, Aleš Horák (Eds.): Proceedings of Recent Advances in Slavonic Natural Language Processing, RASLAN 2009, pp. 49–54, 2009.

In the next section, we describe the linguistic pre-processing yielding the underlying linguistic analysis. Section 3 sketches the architecture of the Saara framework performing AR itself, and next, Section 4 addresses the issues relevant to the synthesis of all the modules mentioned. Finally, we present a summary of the work presented and discuss directions of future work.

2 Syntactic Analysis

Like any higher-level linguistic processing of texts, anaphora resolution within our system requires support in lower-level analysis – especially in information about morphological and syntactic structure.

As the first step, the input text is tokenized and further processed by a morphological tagger, which carries out automatic morphological analysis and disambiguation. The tools used have been developed at the NLP Center, MU Brno. Notably, the morphological tagger is based on the moprhological analyzer ajka [4] and the chunk parser VaDis [5].

The subsequent syntacic analysis is performed using the "synt" syntactic analyzer developed by at the NLP Laboratory, FI MU, Brno [6]. The parsing is carried out in a head-driven chart-parse manner with context-free rules defined in three forms: G1, G2, and G3. G1 is a meta-grammar edited by human experts, mainly taking care of the combination of phrases, especially verbal ones. The Second Grammar Form, G2 contains also a description of context actions associated with individual G1 grammar rules. These prune combinations where conditions on agreement in grammatical categories are not met. The Expanded Grammar Form, G3 contains all necessary feature agreement tests as context-free rules.

The whole process yields a number of most probable phrasal derivation trees. These are selected and ordered using statistics concerning probability of the individual analyses and semantic features, such as verb valencies.

The following section describes how this computed structure is further utilized to reach information about anaphoric relations.

3 The Saara Framework

At present, mechanisms for performing anaphora resolution are becoming integral parts of modern NLP systems. Disregarding anaphora resolution inevitably means creating a serious bottleneck within the linguistic analysis process.

For Czech, various AR algorithms have been proposed (e.g. [7,8,9]), however, due to the inavailability of suitable Czech linguistic resources, haven't been implemented. The emergence of the Prague Dependency TreeBank [2,3], which contains annotation of pronominal coreference led to the occurrence of two AR systems: the above-mentioned AČA system by Linh [1], and the Saara Framework [10] presented below.

The architecture of the Saara Framework has been greatly inpired by earlier AR systems, especially the one developed by Byron and Tetreault [11] at the University of Rochester. They emphasise the advantages of modularity and encapsulation of the system modules into layers. Themselves, they propose three layers:

- the AR layer containing functions addressing AR itself,
- the translation layer for creating data structures,
- the supervisor layer for controlling the previous layers.

The Saara Framework exhibits a very similar distinction of processing layers. There is the so-called *"markable layer"* which is used to define the actual AR algorithms. The main feature of this layer is its maximal generality. It has access only to a general discourse model consisting of the basic discourse structure, the so-called markables, representing discourse objects and a limited number of interface functions describing relationships between them. Next, there is the *"technical layer"* which describes the actual representation of the text, in the particular formalism and format used. Furher, it encompasses the implementations of the functions from the markable layer, translating their abstract idea into the terms of the formalism in question. These two layers correspond to the first two layers mentioned by Byron and Tetreault. Their *"supervisor layer"* can be thought of in Saara as of a layer of very short programs that define a sequence of pre-processing and markable-layer modules to be called, with the specification of their parameters.

The markable-layer modules, that is AR algoritms, re-implemented and available in the Saara framework, are mainly traditional algorithm based on modeling of salience:

Plain Recency is a baseline algorithm linking each anaphor to the closest antecedent candidate agreeing in morphology.

The Hobbs' Syntactic Search [12] is one of the earliest AR approaches and unlike the other algorithms mentioned here, it is furmulated as a search by traversing the syntactic trees representing the discourse.

The BFP Algorithm [13] is based on the principles of Centering theory. It models local coherence among utterances and uses this concept to suggest anaphoric links resulting in the most coherent discourse.

Activation models considering TFA[1] [7,9] have been formulated within the Praguian framework of Functional Generative Description and are based on modeling the level of activation the individual discourse objects have in the mind of the reader.

The method of combining salience factors inspired by the RAP system by Lappin and Leass [14] is based on specifying various factors that favour (or disfavour) individual referential expressions as antecedents for anaphors. Assigning appropriate weights to individual factors allows very flexible modeling of salience.

[1] TFA stands for Topic-focus articulation; similar ideas are also known as information structure, or functional sentence perspective.

Performance of AR algorithms is very difficult to evaluate. A number of metrics have been proposed to assess the correctness of AR systems numerically, however, there is a broad range of factors that bias these numbers considerably: whether errors propagated from the pre-processing are counted, whether AR is carried out on a pre-defined set of markables or the errors in detecting anaphoric and non-referential expressions are included, the precise types of anaphora addressed, genre of the text etc. For these reasons, the figures in Table 1 are given only for the purpose of comparing the individual algorithms within our framework (not our system with other systems), revealing their advantages and disandvanteges when used on same texts.

Table 1. Performance of the system in MUC-6 measures

	Precision	Recall
Plain Recency	41.78	37.28
Hajičová 1987	41.33	36.81
Hajičová, Hoskovec, Sgall, 1995	41.33	36.80
Hobbs' syntactic search	38.87	33.91
BFP Centering	52.26	39.20
Lappin and Leass' RAP	49.86	46.28

4 Anaphora Resolution over Pre-parsed Czech Text

The two preceding sections have described the application setting used to perform automatic AR over previously unprocessed Czech text.

For each sentence, the pre-processing phase yields an ordered sequence of trees given in a bracket notation – each node representing either a terminal or non-terminal phrasal node, carrying information about its morphological features and syntactic category. As the trees are sorted according to their estimated plausibility, further processing takes advantage of the first one only.

For the AR algorithms to function correctly, we need to determine certain important structures within the derivational trees of the individual sentences.

Firstly, each sentence needs to be divided into clauses. This can be done straightforwardly based on the syntactic category tags provided by the "synt" parser. Clauses are crucial in the next step of the processing, the detection of zero subjects.

Czech is a pro-drop language, meaning that subjects of clauses need not necessarily be realized phonologically. Such, so-called, zero subjects do not correspond to any token within the text and thus are obviously missing from the syntactic parse of the sentence. They need to be added, as they play a key role in textual anaphoric relations. When a sentence does not contain any nominal phrase in nominative (and does not contain a subjectless verb), a subject node is added to the beginning of the sentence, with morphological features determined based on the verbal complex of the sentence.

As a next step, referential expressions are detected within the text, based on the syntactic category tags given by the parser. This already compiles a substantial part of the discourse model allowing the AR procedures to process the text. The only necessary issue left is to define an interface between abstract phenomena considered within the AR process, and their actual representation in the given formalism. This mainly concerns determining individual syntactic roles and ordering of referential expressions within clauses. Within "synt" derivational trees, this is done heuristically using morphological features of phrases and their linear order within the sentence.

After AR is carried out using the chosen algorithms (resolution of grammatical and textual anaphora is performed separately, one after the other), the Saara framework yields a set of markables divided into equivalence classes that are induced by coreference (or other anaphoric relation in question). This data is exported into the MMAX2 XML format for the purposes of visualisation and further processing.

MMAX2 [15] is an annotation tool that can be used to store and display data of various kinds, and to annotate various phenomena in them. The annotation can be multi-layer, which means that one annotation project can encompass a number of different unrelated phenomena, or a sequence of mutually dependent ones. For AR data, we define three separate layers over text tokens:

- sentences
- clauses
- referential expressions (grouped into coreferential sets)

Each of these annotation layers computed by the Saara framework algorithms are stored within an XML file with a straightforward structure, and can be easily used for further processing.

5 Conclusions and Further Work

This article has presented a number of linguistic tools developed at the NLP Center, MU Brno, namely the "synt" syntactic analyzer and the Saara framework for automatic AR. We mentioned how the syntesis of these tools is used to carry out AR over previously unprocessed Czech texts and discussed a number of interesting issues within this process.

Our further work aims at improving the accuracy of detecting syntactic structures. This can be done by considering more shallow structures that can be computed with stronger reliability. Further, we plan to enhance the AR algorithms themselves, by employing various semantic features, such as WordNet-like semantic classes or valency data. We also plan to use Saara with English texts.

Acknowledgments This work has been partly supported by the Ministry of Education of CR within the Center of basic research LC536.

References

1. Linh, N.G.: Návrh souboru pravidel pro analýzu anafor v českém jazyce. Master's thesis, Charles University, Faculty of Mathematics and Physics, Prague (2006).
2. Hajič, J., et al.: The Prague Dependency Treebank 2.0. Developed at the Institute of Formal and Applied Linguistics, Charles University in Prague. (2005) `http://ufal.mff.cuni.cz/pdt2.0/`.
3. Kučová, L., Kolářová, V., Žabokrtský, Z., Pajas, P., Čulo, O.: Anotování koreference v pražském závislostním korpusu. Technical report, Charles University, Prague (2003).
4. Sedláček, R.: Morfologický analyzátor (čestiny). Ph.D. thesis, Fakulta informatiky Masarykovy univerzity v Brně, Brno (1999).
5. Žáčková, E.: Parciální syntaktická analýza (čestiny). Phi.D. thesis, Fakulta informatiky Masarykovy univerzity v Brně, Brno (2002)
6. Horák, A.: Computer Processing of Czech Syntax and Semantics. Librix.eu, Brno, Czech Republic (2008).
7. Hajičová, E.: Focusing – a meeting point of linguistics and artificial intelligence. In Jorrand, P., Sgurev, V., eds.: Artificial Intelligence Vol. II: Methodology, Systems, Applications. Elsevier Science Publishers, Amsterdam (1987) 311–321.
8. Hajičová, E., Kuboň, P., Kuboň, V.: Hierarchy of salience and discourse analysis and production. In: Proceedings of Coling '90, Helsinki (1990).
9. Hajičová, E., Hoskovec, T., Sgall, P.: Discourse modelling based on hierarchy of salience. The Prague Bulletin of Mathematical Linguistics (64) (1995) 5–24.
10. Němčík, V.: The Saara Framework – work in progress. In: RASLAN 2008, Recent Advances in Slavonic Natural Language Processing, Brno (2008) 11–16.
11. Byron, D.K., Tetreault, J.R.: A flexible architecture for reference resolution. In: Proceedings of the Ninth Conference of the European Chapter of the Association for Computational Linguistics (EACL '99). (1999).
12. Hobbs, J.R.: Resolving pronoun references. In: Grosz, B.J., Spärck-Jones, K., Webber, B.L., (eds.): Readings in Natural Language Processing. Morgan Kaufmann Publishers, Los Altos (1978) 339–352
13. Brennan, S.E., Friedman, M.W., rd, C.J.P.: A centering approach to pronouns. In: Proceedings of the th25 Annual Meeting of the ACL, Stanford (1987) 155–162.
14. Lappin, S., Leass, H.J.: An algorithm for pronominal anaphora resolution. Computational Linguistics **20**(4) (1994) 535–561.
15. Müller, C., Strube, M.: Multi-level annotation of linguistic data with MMAX2. In Braun, S., Kohn, K., Mukherjee, J., eds.: Corpus Technology and Language Pedagogy: New Resources, New Tools, New Methods. Peter Lang, Frankfurt a.M., Germany (2006) 197–214.