Mining Lemma Disambiguation Rules from Czech Corpora

Luboš Popelínský and Tomáš Pavelek

Natural Language Processing Laboratory
Faculty of Informatics, Masaryk University in Brno
Czech Republic
{popel,xpavelek}@fi.muni.cz

Abstract. Lemma disambiguation means finding a basic word form, typically nominative singular for nouns or infinitive for verbs. In Czech corpora it was observed that 10% of word positions have at least 2 lemmata. We developed a method for lemma disambiguation when no expert domain knowledge is available based on combination of ILP and kNN techniques. We propose a way how to use lemma disambiguation rules learned with ILP system Progol to minimise a number of incorrectly disambiguated words. We present results of the most important subtasks of lemma disambiguation for Czech. Although no knowledge on Czech grammar has been used the accuracy reaches 93% with a small fraction of words remaining ambiguous.

1 Disambiguation in Czech

Disambiguation in inflective languages, of which Czech is a very good instance, is a very challenging task because of their usefulness as well as its complexity. DESAM, a corpus of Czech newspaper texts that is now being built at Faculty of Informatics, Masaryk University, contains more than 1 000 000 word positions, about 130 000 different word forms, about 65 000 of them occuring more then once, and 1665 different tags.

DESAM is now being tagged – partially manually, partially by means of different disambiguators – into 66 grammatical categories like a part-of-speech, gender, case, number etc., about 2 000 tags, combinations of category-value couples. E.g. for substantives, adjectives and numerals there are 4 basic grammatical categories. For pronouns 5 categories, for verbs 7 and for adverbs 3 categories, and some number of subcategories. The large number of tags is made by combination of those categories. It was observed [11] that there is in average 4.21 possible tags per word. It is impossible to perform the disambiguation task manually and any tool that can decrease the amount of human work is welcome.

DESAM is still not large enough. It does not contain all Czech word forms – compare 132 000 different word forms in DESAM with more than 160 000 stems of Czech words that morphological analysers are able to recognise (each of them can have a number of both prefixes and suffixes). Thus DESAM does not

J.M. Żytkow and J. Rauch (Eds.): PKDD'99, LNAI 1704, pp. 498–503, 1999.

contain the representative set of Czech sentences. In addition DESAM contains some errors, i.e. incorrectly tagged words. Another problem is that the significant amount of word positions (words as well as interpunction) are untagged. For the word form “se” nearly one fifth of words are untagged (16,8%) and 93.4% of contexts contain an untagged word. It is similar for other classes of words with an ambigoues lemma.

It should be noticed here that the disambiguation task in Czech language is much more complex than in e.g. English also for another reason. For English there are tagged corpora covering a majority of common English sentences. The known grammar rules cover a significant part of English sentence syntax. Unfortunately, neither of those statements hold for Czech. It makes our task quite difficult.

2 Lemma Disambiguation

Lemma disambiguation which we address here, means assigning to each word form its basic form – nominative singular for nouns, adjectives, pronouns and numerals, infinitive for verbs. E.g. in the sentence *Od rána je má Ivana se ženou.* (literarily *since (the) morning my Ivana(female) has been with (my) wife.*) each of words except the preposition ”od” has two basic forms. E.g. “rána” can be genitive of “ráno”(morning) as well as nominative of a substantive “rána”(bang).

In Czech corpora it was observed that 10% of word positions – i.e. each 10th word of a text – have at least 2 lemmata and about 1% word forms of the Czech vocabulary has at least 2 lemmata. The most frequent ambiguous word forms are **se** and **je**. Disambiguation of the word “se” would be welcome as it is the 3rd most frequent word in DESAM corpus. Actually the lemma disambiguation (almost always) leads to a disambiguation of sense. In the example, *má* means either ”my” (daughter) or ”has” (s/he has), *se* is either preposition ”with” (my wife) or the reflexive pronoun “self” (like ”elle *se* lave” in French).

We use here a novel approach to lemma disambiguation based on a combination of memory-based learning, namely weighted k-Nearest Neighbor method [8] and inductive logic programming(ILP) [9]. Inductive logic programming aims at finding first-order logic rules that cover positive examples (and uncover negative ones) using given domain knowledge predicates.

■

The rest of the paper is organised as follows. In Section 3 we explain how to build basic domain knowledge if no sufficient linguistic knowledge is available. In the Section 4 we present the results obtained with ILP system Progol for the most frequent lemma-ambiguous word form “se”. Rule set accuracy on a disambiguated context is displayed. Section 5 brings the results of disambiguation when correct tags in a context are unknown. We conclude with a discussion of results and a summary of relevant works.

3 Domain Knowledge

There is no complete formal description of Czech grammar. We decided to build domain knowledge predicates without any need of deep linguistic knowledge. We only exploit information about particular tags in a context. The general form of domain knowledge predicates is

```
p(Context, Focus, Condition)
```

where `Context` is a variable bound with either left context in a reverse order or with right context, `Focus`, `Condition` are terms. `Focus` defines a subpart of the `Context`. It has a form `first(N)` (N=1..max_length, a sublist of the `Context` of length `N` neighboring with the word. `max_length` is a maximal length of a context). `Condition` says what condition must hold on the `Focus`. `Condition` is an unary term of the form `somewhere(List)` (tags from the `List` appear somewhere in the `Context`) or `always(List)` (tags from the `List` appear in all positions in the `Context`). E.g. a goal

```
p(X,first(2), always([c7,nS]))
```

succeeds if tags `c7,nS` appear in each of the first two words in the context `X` – e.g. a pronoun and a noun in singular instrumental as in "(se) svou sestrou" – "(with) his sister".

4 Learning Disambiguation Rules with Progol

We will demonstrate our method on disambiguation of the word form "se". It may have either the lemma "s" (preposition like "with" in English) or the lemma "sebe" (reflexive pronoun "self").

For generation of learning sets we use the part of DESAM corpus which was manually disambiguated (about 250 000 word positions). The left and right contexts have been set to 5 words. Untagged words in context has been tagged as 'unknown part-of-speech' (tag `kZ`). Negative examples have been built from sentences where the word has the second lemma.

Using P-Progol [10] version 2.2 we have learned rules for both of the two lemmata. It means that for each task we obtained two rule sets that should be complementary. However, we have found it useful to use both of them. Number of sentences was 232 (preposition) and 2935(pronoun). 80% of examples was used for learning. We tested each rule set on the rest of data. Learning time reached 14 hours. It is caused by the enormous number of 4536 literals that may appear in a rule body. It must be mentioned that the default accuracy, i.e. assigning the reflexive pronoun lemma to each occurrence of "se" is 92.7%. Then the reached rule accuracies 92.84% (pronoun) and 94.48%(preposition) are not too impressive. In the next section we will show that even such "poor" rule sets are usable for lemma disambiguation.

5 Disambiguation

The goal then was to find such a criterion that would allow to find the correct lemma for the word "se". The learning and the testing sets contained sentences not used for learning the disambiguation rules. We limited both left and right context to the length of 3 words. Then we removed all sentences that contained commas, dots, parentheses etc. 50% of the sentences were used for estimation of parameters, the rest for testing. All possible grammatical categories were found for each sentence employing LEMMA morphological analyser[1]. Then all variations of categories was generated for each sentence. Both theories learned by Progol were run on those data so that for each sentence we had two success rates, i.e. the relative number of correctly covered positive examples and correctly uncovered negative examples to the number of all examples. Time needed for disambiguation of 1 sentence was 6 seconds in average, very rarely it was more than 10 seconds. If the disambiguation lasted more than 30 seconds (because of the enormous number of variations of tags), the process was killed. It concerned less than 2% of cases.

Two success rates obtained for a sentence are as (x,y)-coordinates. The new example is then classified into class(lemma) of the nearest neighbor(s) in that learning set. We computed the distance between two instances (x_1, y_1) and (x_2, y_2) as an Euclidian distance. As mentioned above, 50% new sentences have been used for building the set of instances and for parameter estimation. On the new learning set we tried values of k (the number of neighbors) in the range 1..10. It was observed that increasing value of k did not increase accuracy of disambiguation. Therefore for all experiments below k was set to 1. Then we found the nearest point (x_i, y_i). Let s_1, s_2 be the number of instances with lemma "s" and the number of instances with the lemma "sebe" for this point. If s_i is greater than s_j we would expect that the i-th lemma is the right one. We also observed that if a success rate for a word in a particular context is smaller than a threshold the word cannot be disambiguated. Thus the correct lemma was assigned using the rules in Fig. 1. Values of $(t_{lemma_1}, t_{lemma_2})$ was tested in the range (0,0)..(1,1). The best settings of thresholds on the learning set was $t_{lemma_1} = 0,\ t_{lemma_2} = 0.8$. Results of disambiguation are in Table 1.

```
lemma := if $s_1 > s_2 \ \wedge succesRate_{lemma_1} > t_{lemma_1}$ then $lemma_1$
         else if $s_1 < s_2 \ \wedge succesRate_{lemma_2} > t_{lemma_2}$ then $lemma_2$
         else $unresolved$
```

Fig. 1. kNN algorithm

[1] copyright Lingea Brno 1995

			disambiguation			unresolved	
		#ex	correct	wrong	accuracy(%)	#	%
preposition	learn	99	80	4	97.5	17	17.2
	test	112	93	7	93.0	14	12.5
pronoun	learn	297	214	2	99.1	82	27.6
	test	310	236	6	97.5	44	14.2

Table 1. Results of kNN algorithm

6 Conclusion

The presented results are the first obtained by ILP techniques in disambiguation of inflective languages, as far as we know. It must be stressed that the Czech corpus is under development and therefore it contains about 17% of untagged words as well as incorrectly tagged words. Moreover, there were no usable formal grammar rules for Czech that would make the domain knowledge building easier. We described the systematic way of building domain knowledge if no sufficient linguistic knowledge is available. A new method for lemma disambiguation was introduced that reached an accuracy 93%, leaving a small part of words ambiguous. Similar accuracy was obtained for Prague Tree Bank corpus [13]. The lemma disambiguation task is not solved here completely. The main reason is that the Czech corpora are still too small and therefore cardinality of learning sets is not sufficient for most of the tasks.

Our approach was also used for disambiguation of unknown words (not existing in the corpus). We defined similarity classes for lemma-ambiguous words in terms of grammatical categories. First results can be found in [12]. Results obtained by ILP for tag disambiguation can be found in [13].

So far statistical techniques (accuracy 81.64%) and neural nets (75.47%) have been applied to DESAM [11]. See also [5,6,14] for other results with another Czech corpus. It should be pointed out that our results are not quite comparable as we focus only on lemma disambiguation.

In the past, ILP has been applied for inflective languages in the field of morphology. LAI Ljubljana [2] applied ILP for generating the lemma from the oblique form of nouns as well as for generating the correct oblique form from the lemma, with average accuracy 91.5 % . Learning nominal inflections for Czech and Slovene (among others) is described in [7]. James Cussens [1] developed POS tagger for English that achieved per-word accuracy of 96.4 %. Martin Eineborg and Nikolaj Lindberg [3,4] induced constraint grammar-like disambiguation rules for Swedish with accuracy 98%. Our approach differs significantly in two points. We do not exploit any information on particular words as in [3]. Such knowledge would improve an accuracy significantly. Neither we use any hand-coded grammatical domain knowledge as in [1].

Out method, although developed for Czech language, is actually language-independent except of the set of tags. It means that it is possible to use our approach also for other languages.

Acknowledgements

This paper is a brief version of [13]. We thank a lot to anonymous referees of LLL workshop for their comments. We would like to thank to Karel Pala and Olga Štěpánková for their help with earlier versions of this paper. We thank, too, Tomáš Ptáčník, Pavel Rychlý, Radek Sedláček and Robert Král for fruitful discussions and assistance. This work has been partially supported by VS97028 grant of Ministry of Education of the Czech Republic "Natural Language Processing Laboratory" and ESPRIT ILP^2 Project.

References

1. Cussens J.: Part-of-Speech Tagging using Progol. In Proc. of ILP'97, LNAI 1297, Springer-Verlag 1997.
2. Džeroski S., Erjavec T.: Induction of Slovene Nominal Paradigms. In Proc. of ILP'97, LNAI 1297, Springer-Verlag 1997.
3. Eineborg M., Lindberg N.: Induction of Constraint Grammar rules using Progol. In Proc. of ILP'98, 1998.
4. Lindberg N., Eineborg M.: Learning Constraint Grammar-style disambiguation rules using Inductive Logic Programming. In: COLING/ACL'98.
5. Hajič J., Hladká B.: Probabilistic and rule-based tagger of an inflective language – a comparison. In Proceedings of the 5th Conf. on Applied Natural Language Processing, 111-118, Washington D.C., 1997.
6. Hajič J., Hladká B.: Tagging Inflective Languages: Prediction of Morphological Categories for a Rich, Structured Tagset. In Proceedings of EACL 1998.
7. Manandhar S., Džeroski S., Erjavec T.: Learning multilingual morphology with CLOG. In Proc. of ILP'98, 1998.
8. Mitchell, T.M.: Machine Learning. McGraw Hill, Newy York, 1997.
9. Muggleton S., De Raedt L.: Inductive Logic Programming: Theory And Methods. J. Logic Programming 1994:19,20:629-679.
10. Muggleton S.: Inverse Entailment and Progol. New Generation Computing Journal, 13:245-286, 1995.
11. K. Pala , P. Rychlý and P. Smrž: DESAM - annotated corpus for Czech. In Plášil F., Jeffery K.G.(eds.): Proceedings of SOFSEM'97, Milovy, Czech Republic. LNCS 1338, Springer-Verlag 1997.
12. Pavelek T., Popelínský L.: Towards lemma disambiguation: Similarity classes (submitted)
13. Popelínský L., Pavelek T., Ptáčník T.: Towards disambiguation in Czech corpora. Workshop Notes of "Learning Language in Logic"(LLL) ICML'99 Workshop, Bled, Slovenija, 1999.
14. Zavrel J., Daelemans W.: Recent Advances in Memory-Based Part-of-Speech Tagging. ILK/Computaional Linguistics, Tilburg University, 1998.