Data-Driven Methods for PoS Tagging and Chunking of Swedish

Beáta Megyesi
Centre for Speech Technology
Royal Institute of Technology
SE-100 44, Stockholm, Sweden
bea@speech.kth.se

Abstract

In this paper well-known state-of-the-art data-driven algorithms are applied to part-of-speech tagging and shallow parsing of Swedish texts.

1 Introduction

In recent years, machine learning has become very popular for natural language processing (NLP) tasks, such as part of speech (PoS) tagging and shallow parsing, because the algorithms automatically and efficiently can learn from natural language data given a correctly annotated training corpus. There is a vast number of algorithms that have been developed and applied with good results to analyze natural languages on different linguistic levels. For example, Hidden Markov Modeling (Brants, 2000), Maximum Entropy (Ratnaparkhi, 1996), Memory-Based Learning (Zavrel & Daelemans, 1999) and Transformation-Based Learning (Brill, 1994) have been successfully applied to PoS tagging of English with an average accuracy of between 95% and 97%. Recently some attempts also have been made to build data-driven shallow parsers for English by finding syntactically related non-overlapping group of words, so called chunks (Abney, 1991)[1].

In this study, data-driven algorithms are applied to PoS tagging and chunking of Swedish. Common to these algorithms is that they have implementation for PoS tagging, are claimed to be language- and tag set-independent, and easily applicable to new languages given a set of correctly annotated training data.

First, each algorithm will be briefly described. Second, the evaluation and comparison of the data-driven PoS taggers is presented. Lastly, the method used for building shallow parsers and the results are given.

2 Data-Driven learning algorithms

Four well-known data-driven algorithms are used to analyze Swedish texts on different linguistic levels. Each algorithm is briefly described below.

Memory-Based Learning (mb), described by Daelemans et al. (1996), is a case-based approach where new items are classified on the basis of similarities to the earlier examples stored in memory during learning. In this study, decision tree induction, called ig-tree, was re-implemented for Swedish by Harald Berthelsen, based on the description given in Zavrel & Daelemans (1999)[2]. Here, an instance is represented by a vector where the elements are the different features of the instance. The system contains information about the focus word, the preceding and following word forms, the two preceding tags and the one following tag for known words. For unknown words, information about capitalization, the presence of a hyphen or a numeral feature, the preceding tag, the focus word, the ambiguous right tag and the last three letters occurring in the word is used.

The Maximum Entropy (me) framework, called mxpost, is described by Ratnaparkhi (1996). It is a probabilistic classification-based approach based on a Maximum Entropy model where contextual information is represented as binary features that are used simultaneously in order to predict the PoS tag. The default binary features include the current word, the following and preceding two words and the preceding

[1] For a sample of individual research efforts in the field of data-driven chunking, see the Proceedings of the 4th Conference on Computational Natural Language Learning, 2000.

[2] The re-implementation of the Swedish tagger was necessary because it was not available on the ILK web page (http://ilk.kub.nl/software.html).

two tags. For rare and unknown words the first and last four characters are included in the features, as well as information about whether the word contains uppercase characters, hyphens or numbers. The tagger uses a beam search in order to find the most probable sequence of tags. For known words it generates the possible tags, and for unknown words it generates all tags in the tag set. The tag sequence with the highest probability is chosen.

TRANSFORMATION-BASED LEARNING (TBL), developed by Brill (1995), is a rule-based approach that learns by detecting errors. It begins with an unannotated text that is labeled by an initial-state annotator in a heuristic fashion. Then, an ordered list of rules learned during training is applied deterministically to change the tags of the words according to their contexts. TBL uses a context of three preceding and following words and/or tags of the focus word. Unknown words are first assumed to be nouns and handled by prefix and suffix analysis by looking at the first/last one to four letters, capitalization feature and adjacent word co-occurrence.

TRIGRAMS'N'TAGS (TNT) is a statistical approach, developed by Brants (2000). The tagger is a trigram Hidden Markov Model and uses the Viterbi algorithm with beam search for fast processing. The states represent tags and the transition probabilities depend on pairs of tags. The system uses maximum likelihood probabilities derived from the relative frequencies. The main smoothing technique implemented by default is linear interpolation. Unknown words are handled by suffix analysis, i.e. up to the last ten letters of the word. Additionally, information about capitalization is included as default.

3 Data-driven PoS taggers

The algorithms applied to annotate Swedish texts with PoS and morphological features are the Maximum Entropy approach (ME) (Ratnaparkhi, 1996), Memory-Based Learning (MB) (Daelemans, et al., 1996), Transformation-Based Learning (TBL) (Brill, 1994), and Trigrams'n'Tags, (TNT) (Brants, 2000). The aim is to find out how well the algorithms are able to annotate Swedish with PoS and morphological features, to find out the advantages and drawbacks of the methods and to describe the type of errors they make, the effects of the tag set size, and the effect of the size of training material.

Table 1: The tagging accuracy for all the words, and the accuracy of known and unknown words are given for each classifier. Training and test set are disjoint, consisting of 100k tokens, respectively. Tag set includes 139 tags.

ACCURACY	MB	ME	TBL	TNT
TOTAL %	89.28	91.20	89.06	**93.55**
KNOWN %	92.85	93.34	94.35	**95.50**
UNKNOWN %	68.65	78.85	58.52	**82.29**

All experiments were run on the second version of Stockholm-Umeå Corpus (SUC), annotated with Parole tags (Ejerhed, et al., 1992). The SUC corpus was randomly divided into ten approximately equal parts in order to get subsets containing different genres. For a fair comparison of the methods, each algorithm was trained in each experiment on the same part of the SUC corpus to build four classifiers. Then, each classifier was evaluated on the same test set (117685 tokens) of which 85.23% are known and 14.77% are unknown words. The training and the test set were disjoint. The classifiers were allowed to assign exactly one tag to each token in the test. The baseline performance is 77.37% and is obtained on the test data by selecting the PoS tag that is most frequently associated with the current word. The systems were evaluated from three different aspects.

First, the average accuracy was counted for each classifier, trained on 10% of SUC (115862 tokens) with the entire tag set consisting of 139 tags. The results, given in Table 1, show that all systems outperformed the baseline, but the performance of the taggers is significantly lower than is reported for English.

TnT has the highest overall accuracy and also succeeds best in the annotations of known and unknown words. The ME tagger shows high performance because of the high precision of the annotation of unknown words. TBL manage to disambiguate known words but succeeds poorly on unknown words. MB is slightly better than TBL because of its better success in the annotation of unknown words. Furthermore, TBL and MB more often make mistakes in the mor-

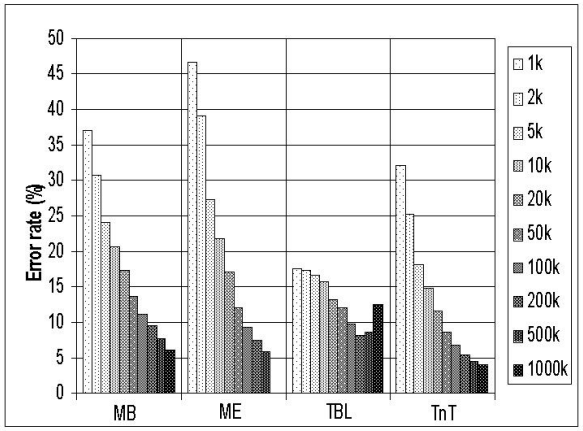

Figure 1: Error rates when training on 1000 to 1 million tokens, totally ten training corpora of various sizes, seen as ten columns for each classifier.

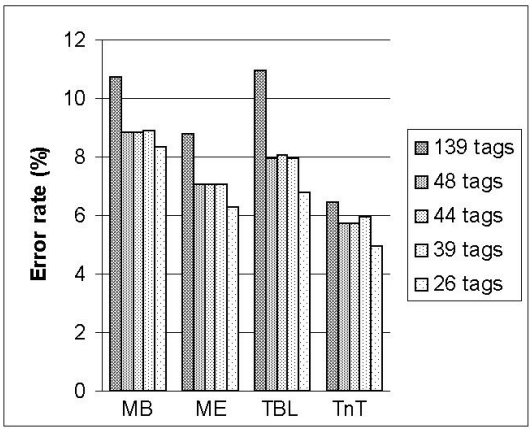

Figure 2: The error rate for each classifier when training on 139, 48, 44, 39 and 26 tags.

phological analysis of categories while ME and TNT more frequently confuse ambiguity classes among PoS categories.

Secondly, each algorithm was trained ten times on the same data set of various size from one thousand to one million tokens. Then, the same test set was annotated by each classifier. The results (see Figure 1) show that larger training data improves the overall accuracy greatly for MB, ME and TnT, but not for TBL. The reason for the low error rate of TBL is the possibility to use a large lexicon which decreases the amount of unknown words and increases the amount of possible categories for each token.

Lastly, each algorithm was trained on different size of tag sets: 139 tags, 48, 44, 39 and 26 tags. The results are shown in Figure 2. By decreasing the size of the tag set from 139 to 26 tags, the error rate decreases by 38% for TBL, 29% for ME, and 23% for MB and TNT. Thus, TBL and ME seem to be more sensitive to the size of tag set than MB and TNT. Furthermore, training on between 39 and 48 tags, the system performances show rather similar results. Thus, the size of the tag set as well as the type of information are crucial facts for system performance.

Concluding the results, TNT has the highest overall accuracy, succeeds best in the annotation of known as well as unknown words, and also fastest in both training and tagging. TBL has high performance on small training corpora, hence it can be used as an aid when building large corpora by applying a boot-strapping procedure. ME has high error rate of the annotation of known tokens when training on small corpora. MB is fast in both training and test and succeeds better in the morphological disambiguation than ME.

4 Data-driven chunkers/parsers

The purpose of this part of the study is to build data-driven shallow parsers by using the chunk-tag technique, i.e. divide the text into syntactically related non-overlapping groups of words, phrases. Thus, the aim is not the disambiguation of words according to their context since disambiguation takes place on the level of PoS annotation instead with the help of some background knowledge implemented in the PoS taggers containing information about contextual environment of the current word or tag.

The facts that the data-driven PoS taggers have knowledge about the contextual environment and are language- and tag set-independent lead to the thought that they can be assumed to be useful to parse texts, given a correctly annotated training data.

Since correctly chunked/parsed texts are not available for Swedish, a tree-bank was built to serve as training data and bench-mark corpus. For this purpose, an Earley Parser, SPARK (Aycock, 1998) together with a context-free grammar for Swedish developed by the author, was used.

The second version of the Stockholm-Umeå

corpus annotated with PAROLE tags served as input to the parser. The PoS tagged texts were parsed by SPARK for Swedish to serve as training data and bench-mark corpus.

Nine types of phrases were included: adverb phrase (ADVP), minimal adjective phrase (APMIN), maximal adjective phrase (APMAX), noun phrase (NP), preposition phrase (PP), maximal projection of NP (NPMAX), verb clusters (VC), infinitive phrase (INFP) and numeral expression (NUMP).

Additionally, each chunk was represented as three types of tags in a similar way as it was proposed by Ramshaw (1995) and used in the CoNLL-2000 competition:

– XB - the first word of the chunk X
– XI - non-initial word inside the chunk X
– O - word outside of any chunk.

Each word and punctuation mark in a sentence is accompanied by a tag which indicates the phrase structure the word belongs to in the parse tree together with the position information. Thus, a word may belong to several phrases as illustrated in the example below for the sentence 'The review of papers should be blind', represented first by parenthesis, and second by tags.

[NPMAX [NP Granskningen NP] [PP av [NP artiklar NP] PP] NPMAX] [VC borde vara VC] [AP anonym AP].

Granskningen/NPB_NPMAXB
av/PPB_NPMAXI
artiklar/NPB_PPI_NPMAXI
borde/VCB
vara/VCI
anonym/APMINB
./0

Thus, the label for a word forms a hierarchical grouping of the parts of the sentence into constituents where lower nodes are situated nearest the word and higher nodes are farthest out. The advantage of the hierarchical annotation on phrase level is that the user can choose the level of the analysis by skipping phrase categories on lower, or higher nodes. For example, the user may only want to use noun phrase extraction without any information on the constituents inside the noun phrase, or to get full analysis of every large phrase in the sentence. This type of annotation can be used in many different applications.

Three data-driven algorithms that have implementations for the PoS tagging approach are applied to build data-driven chunkers: MXPOST, based on the Maximum Entropy framework, Transformation-Based Learning (TBL), and Trigrams'n'Tags (TNT) based on Hidden Markov Model. The goal is to find out how well the data-driven PoS taggers can learn the hierarchical phrasal structure.

Three types of tests were carried out for each PoS tagger based on the type of linguistic information included in the training data. First, the training corpus contained information about both the word, its PoS tag and the phrase tags. Second, only the word and its phrase tags were included in the training corpus. Third, the words were removed from the training data, only the PoS tags were kept with phrase labels.

In each experiment, the training corpus contained 92109 tokens and the test corpus 23744 tokens. Training and test sets were disjoint and the sentences were randomly chosen from the entire corpus.

The results are shown in the first three raws in Table 2. All systems in all three experiments improved the baseline performance of 59.66%, which was obtained by selecting the phrase tag that is most frequently associated with the current PoS tag.

Large differences can be found in the results depending on the type of information used in training. The systems have lowest performance when the words are annotated with both PoS information and phrase structure information. The low accuracy is not surprising since the tag set consists of a large amount of tags, totally 1033 different combinations of PoS and phrase tags trained on 20946 token (i.e. word) types. When training on both PoS and phrase structure information, the classifiers can be treated as both PoS taggers and parsers.

When PoS information is not present in the training data, the tag set includes 407 different phrase tag combinations. The smaller tag set makes the classification task easier and system performance increases.

Highest accuracy can be obtained when training is done on the basis of PoS and phrase tags only, without the presence of the words. TBL has highest accuracy, followed by ME and TnT. Here, the tag set consists of 407 different phrase

tags – the same tag set that was used in the second experiment but the input (i.e. the PoS tags) to the systems contains only 139 different types. Thus, by decreasing the amount of the type of input data, higher performance can be obtained.

Due to the small size of training and test corpus in the earlier experiments and due to the promising results in the third experiment, the PoS taggers were trained on a larger training corpus using PoS tags and their phrase labels without the inclusion of words, and tested on a larger test set. Totally 139 different PoS tags including morphological features were trained with 570 different types of phrase labels. The training set consisting of 244094 tokens, totally 15640 sentences, and a test set containing 105536 tokens, totally 6698 sentences were used.

As is shown in the last raw of Table 2, accuracy can be further improved by increasing the size of the training corpus, hence increasing the amount of the different contextual environments in which the PoS tag can appear. TBL achieves the highest accuracy, 94.44%, followed by ME and TnT.

Table 2: Accuracy (%) is given for each classifier when training on 92109 tokens and testing on 23744 tokens in three different ways: first the word is annotated with both PoS and phrase tag(s), second, the word is annotated with phrase tag(s) only, and third the PoS served as input and labeled with phrase tags. The last raw shows the accuracy (%) for each classifier when training on 244094 tokens and testing on 105536 tokens using PoS categories labeled with phrase tags.

TYPE OF INFORMATION		ME	TBL	TNT
WORD	POS_PHRASETAGS	**73.78**	68.87	65.36
WORD	PHRASE TAGS	**80.72**	75.47	70.94
POS	PHRASE TAGS	91.58	**92.32**	90.40
POS	PHRASE TAGS	92.47	**94.44**	92.42

5 Conclusions

In this study, state-of-the-art data-driven learning algorithms have been applied to PoS tagging and shallow parsing of Swedish texts.

The first part presented a systematic evaluation and comparison of four data-driven algorithms successfully applied to PoS tagging of Swedish texts with an accuracy up to 97%. The effects of the size of the tag set and the size of the training data have been carefully examined.

The second part presented three state-of-the-art data-driven PoS taggers applied to shallow parse Swedish text. Phrase structure for each token is represented in a hierarchical structure containing tags for every constituent type the token belongs to. The results show that best performance (94.4%) can be obtained by training on the basis of PoS tags with constituent labels without considering the words themselves.

6 References

Abney, S. 1991. Parsing by Chunks. In *Principle-Based Parsing.* Kluwer Academic Publ.

Aycock, J. 1998. Compiling Little Languages in Python. In *Proceedings of 7th International Python Conference.*

Brants, T. 2000. TnT - A Statistical Part-of-Speech Tagger. In *Proceedings of the 6th Applied Natural Language Processing Conference.* Seattle, Washington, USA.

Brill, E. 1994. Some Advances in Rule-Based Part of Speech Tagging. In *Proceedings of the 12th National Conference on Artificial Intelligence (AAAI-94).* Seattle, Washington.

Daelemans, W., Zavrel, J., Berck, P., and Gillis, S.E. 1996. MBT: a Memory-Based Part of Speech Tagger-Generator. In *Proceedings of Fourth Workshop on Very Large Corpora (VLC-96).* pp. 14-27. Copenhagen, Denmark.

Ejerhed, E., Källgren, G., Wennstedt, O., & Åström, M. 1992. *The Linguistic Annotation System of the Stockholm-Umeå Project.* Dept. of General Linguistics, University of Umeå.

Ramshaw, L. A. and Marcus, M. P. 1995. Text Chunking Using Transformation-Based Learning. In *Proceedings of the Third ACL Workshop on Very Large Corpora.* ACL.

Ratnaparkhi, A. 1996. A Maximum Entropy Model for Part-of-Speech Tagging. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP-96).* Philadelphia, PA, USA.

J. Zavrel, and W. Daelemans. 1999. Recent Advances in Memory-Based Part-of-Speech Tagging. In *Proceedings of the VI Simposio Internacional de Comunicacion Social.* Cuba.