Using a morphological analyzer in high precision POS tagging of Hungarian

Péter Halácsy[*], **András Kornai**[**], **Csaba Oravecz**[†], **Viktor Trón**[††], **Dániel Varga**[*]

[*]Media Research and Education Center, Stoczek u. 2, H-1111 Budapest
{hp,daniel}@mokk.bme.hu
[**]MetaCarta Inc., 350 Massachusetts Avenue, Cambridge MA 02139
andras@kornai.com
[†]Research Institute of Linguistics, Benczúr u 33, H-1399 Budapest
oravecz@nytud.hu
[††] University of Edinburgh, 2 Buccleuch Place, EH8 9LW Edinburgh
v.tron@ed.ac.uk

Abstract

The paper presents an evaluation of maxent POS disambiguation systems that incorporate an open source morphological analyzer to constrain the probabilistic models. The experiments show that the best proposed architecture, which is the first application of the maximum entropy framework in a Hungarian NLP task, outperforms comparable state of the art tagging methods and is able to handle out of vocabulary items robustly, allowing for efficient analysis of large (web-based) corpora.

1. Introduction

With trigram-based part of speech (POS) taggers reaching 95-97% correct on Treebank-like English data, POS tagging is viewed by many as a solved problem. Yet there are serious open problems, both for analytic languages like English (where .96 correct tagging means that the average Wall Street Journal sentence of 20 words will be mistagged more often than not) and for highly inflecting languages such as Slovene (Erjavec et al., 1999; Hajič, 2000). Here we focus on the problem from two perspectives: the architecture of the tagging system in terms of the information sources it is designed to utilize, and the limitations of generative models such as the popular TnT (Brants, 2000).

In highly inflecting languages with a large number of possible word forms, if lexical probabilities are calculated from a word form lexicon generated during training, the process will inevitably result in a large number of *unseen* forms in the test data, which degrades the performance of the system (Oravecz and Dienes, 2002). The solution proposed by several authors (Hakkani-Tür et al., 2000; Hajič et al., 2001) is to make full use of existing morphological dictionaries or morphological analyzers (MA) to constrain the probabilistic tagging model and decrease the number of unseen forms. To be sure, there will still be items that are *out of vocabulary* (OOV) for the morphological analyzer, but the proportion if these do not increase with the size of the test data set, while for a fixed training set the proportion of unseens will grow with the size of the testset.

In n-gram models one has to make strict independence assumptions to make the task of sequential data labelling tractable; consequently, long distance dependencies and non-independent features cannot be handled, although they are clearly present in linguistic data. Several answers have been put forward to overcome this limitation in the form of different conditional models, but these have their own problems: maximum entropy or other discriminative Markov models (McCallum et al., 2000) suffer from the label bias problem, while models operating with conditional random fields (CRF) are resource intensive with respect to training, imposing severe limitations on the size of the feature space and training data (Smith et al., 2005).

To cope with these problems we present a hybrid tagging architecture that incorporates a weighted morphological analyzer (WMA) in the maximum entropy framework. The output of the WMA module is pruned by the Viterbi algorithm which operates on a trigram model built during training over possible tag sequences. For Hungarian, the system outperforms all previous taggers, and it offers several advantages over comparable state of the art tagging methods: its critical components are based on open source software (including the morphological analyzer) so it is modifiable and adjustable, it leaves ample room for fine tuning the features it utilizes, and it is robust with respect to OOV items. This last property is especially relevant for the efficient analysis of large (web-based) corpora (Halácsy et al., 2004; Kornai et al., 2006).

The paper is structured as follows. In Section 2. we discuss the difficulty of the labeling task and the baseline use of the MA. Section 3. describes the tagging models based on the maximum entropy framework, and Section 4. presents the result of the evaluation of the methods in several testing scenarios. Section 5. summarizes our conclusions and suggestions for further work.

2. The baseline

The difficulty of morphological disambiguation is generally estimated based on the ratio of ambiguous tokens in the corpus, or on the average number of alternative analyses per token offered by a morphological lexicon. These measures can be significantly distorted by frequent ambiguous tokens: if the lexicon offers alternative analyses, the token is counted as ambiguous irrespective of the probability of the alternatives, even when the selection of the right tag is not problematic for a simple maximum likelihood (ML) estimate.

Thus the difficulty of the task is better measured by the average information required for disambiguating a token. If word w is assigned the label T_i with probability $P(T_i|w)$

(estimated as $C(T_i, w)/C(w)$ from a labelled corpus) then the label entropy for a word can be calculated as $H(w) = -\sum_i P(T_i|w) \log P(T_i|w)$. The difficulty of the labelling task as a whole is the word frequency weighted average of these: $H = \sum_w P(w)H(w)$. For the 1 million word manually annotated Szeged Corpus (Csendes et al., 2004), which we use for test in the experiments, H is 0.1 bits/per word which is considerably lower than the 0.5 bits/word value that would result from taking all alternative analyses equiprobable.

The central problem is that the ratio of unseen items (tokens not seen during the training of the model) has very significant influence on the performance of a disambiguation system. As Oravecz and Dienes (2002) already pointed out, due to very productive morphology, in the same amount of training data (270k words), the ratio of unseen word tokens could be considerably larger in Hungarian (17.1%) than in English (4.5%). To cope with this data sparseness problem three alternative strategies can be followed: (A) increase the size of the training corpus, (B) apply smoothing methods, or (C) use a suitable guesser such as a morphological analyzer to handle unseen words. This last solution is standard, but systems differ greatly in how they utilize the information provided by the MA.

In the following sections we discuss several tagging models that incorporate the same open-source MA called `hunmorph` (Trón et al., 2005), keeping the dictionary (Trón et al., 2006) constant. Our baseline model BMA follows a simple method for using the information from the MA:

(i) If word w is found in the training corpus, BMA will assign the tag for which $T = \text{argmax}_T P(T_i|w)$, otherwise

(ii) if w is known to the MA and gets only one analysis from it, then BMA assigns this tag

(iii) if w is known to the MA but gets multiple analyses, then BMA chooses the one most frequent in the training corpus

(iv) all other tokens are labelled as NOUN.

Since this model ignores contextual information it is not surprising that it will not perform nearly as well as a standard HMM based tagger such as TnT (Brants, 2000), or the combined trigram Markov-model WMA+T3 that we shall describe in the following section. Figure 1 illustrates the learning curve of these three models. Clearly, using MA noticeably improves performance, but without contextual information it is far from the ideal solution (evaluation details are discussed in Section 4.). On the other hand, as training corpus size grows and the ratio of unseen items decreases, the benefits of the information from the MA over the TnT model that uses only a lexicon built from the training corpus become less significant. The main difference between TnT and the WMA+T3 model is that the latter gets the output of the MA for the unseen but not OOV tokens.

Obviously, disambiguation errors are most frequent for tokens that are missing from the training corpus and are at the same time OOV (for the MA). The ratio of these tokens is roughly 2% in the Szeged Corpus. For fixed input text the number of OOV items can be reduced arbitrarily by enhancing the base form lexicon of the MA. For specialized corpora this might be a workable alternative, but as a general solution for large corpora the tagging architecture must be equipped with some module that handles OOVs efficiently without manual extension of the lexicon.

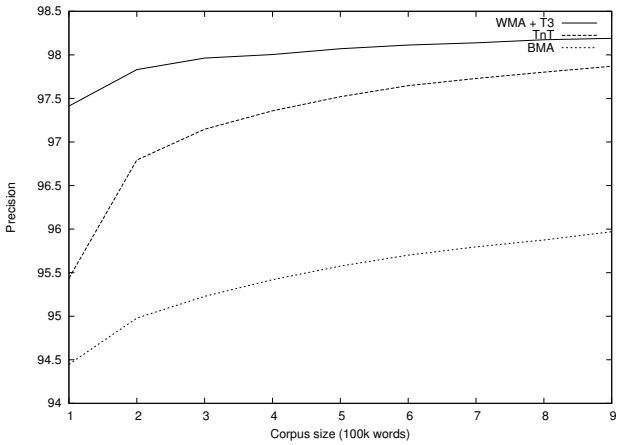

Figure 1: Performance of models as a function of corpus size

3. Maximum entropy based models

Maximum entropy modelling is frequently used in morphosyntactic disambiguation tasks since Ratnaparkhi (1996). For our experiments we rely on the OpenNLP ME library package (Baldridge et al., 2001). To create the model we consider sentences as series of words $w_1, ..., w_n$, for which a corresponding tag sequence $t_1, ..., t_n$ is attributed during training. The ME model defines a joint distribution over possible t_i tags and the actual context c_i,

$$p(t_i, c_i) = \pi \prod_{j=1}^{k} \alpha_j^{f_j(t_i, c_i)} \quad (1)$$

where π is a constant normalisation factor, $\{\alpha_1, ..., \alpha_k\}$ are the model parameters and $\{f_1, ..., f_k\}$ are binary features used in the model. In the disambiguation model (referred to as MA+ME) for the experiments we utilize the following features:

1. the word form in lower case
2. the ambiguity class constructed from the output of the MA for the token
3. the presence of a non-alphabetic characters
4. upper case initial or fully upper case token
5. the last 2,3,4 characters for tokens longer than 5 characters
6. the lower case form of the preceding token for not sentence initial tokens

7. the lower case form of the following token for not sentence final tokens

It is not straightforward to convert the analyses from the MA to features. The best results are obtained when the set of analyses for a token is converted into an ambiguity class and this class is used as one feature in the ME model. Features derived from the word form and its last few characters serve the purpose of handling the OOV items: if the token is unseen and OOV the model can rely only on the features provided by the word ending and the neighbouring words.
In the tagging process a context-specific tag distribution is calculated based on the joint distribution determined by the maxent model. For all words w_i and for all possible tags t_i the following is calculated:

$$P(t_i = T_k|c_i) = \frac{p(t_i = T_k, c_i)}{\sum_{t \in T} p(t_i = T_k, c_i)} \quad (2)$$

The maxent model thus does not make a final selection from the possible tags, only outputs a probability value for each. Although it receives as feature the ambiguity class of the item from the MA, the maxent model assigns positive probability to each tag found in the training corpus.
Our MA+ME model then makes the disambiguation decision according to 2 criteria:

1. If the word is known to the MA then from the set of analyses proposed by the MA the model selects the one that is most probable according to the maxent model. (This subsumes (ii-iii) of the BMA strategy.)

2. If the word is OOV then the maxent model makes the selection.

In terms of possible tag sequences this model backs off to local information just as the preliminary BMA model did: when labelling an item it ignores the tags of neighbouring tokens, in contrast to the HMM based TnT. To overcome this limitation we introduced a hybrid architecture, in which a trigram language model over the tags is combined with the maxent model. Using the maxent model, a weighted morphological analyzer (WMA) is constructed which assigns a probability value to all of its output analyses thus:

1. If the word form is present in the training corpus, tag probabilities are calculated with maximum likelihood estimates just as in the basic models, otherwise

2. If the word form is known to the MA then only the analyses proposed by the MA are allowed as possible output, and their probabilities, as given by the maxent model, are normalized to sum to 1, otherwise

3. For OOVs the 3 most probable tags proposed by the maxent model are considered with probability values normalized to sum to 1.

The WMA thus assigns a set of possible tags to each input token with a corresponding probability value. For the most probable tag sequence over the possible tags we calculate:

$$\underset{t_1,\ldots,t_n}{\operatorname{argmax}} P(t_1, \ldots, t_n|w_1, \ldots, w_n) = \quad (3)$$
$$\underset{t_1,\ldots,t_n}{\operatorname{argmax}} P(w_1, \ldots, w_n|t_1, \ldots, t_n)P(t_1, \ldots, t_n)$$

After standard independence assumptions, the first member of the product is derived from the output of the WMA while the second member is computed on the basis of a trigram language model built on tag sequences from the training corpus. We use the SRILM toolkit (Stolcke, 2002) to build the model and calculate the most probable tag sequence with the Viterbi algorithm. In this model (WMA+T3) the maxent component does not contain features of neighbouring forms to maintain independence among the individual modules. The resulting architecture is very similar in spirit to that proposed by Oravecz and Dienes (2002) although the specific components are different.
Finally, using the above components it is possible to define an architecture similar to Maximum Entropy Markov Models. Here the MA+ME is trained using three additional features, the tag of the preceding/following/actual tokens. During tagging, possible states are the set of analyses the MA allows for known tokens, and all analyses for OOVs, while emission probabilities are estimated by the MA+ME model. In the first pass TnT is run with default settings over the data sequence, and in the second pass the ME receives as features the TnT label of the preceding/following token as well as for the one under scan. This combined 2Pass system incorporates the benefits of all the submodules and achieves the best accuracy.

4. Evaluation

We evaluated the different models by tenfold cross-validation on the Szeged Corpus with separate tests on its five specialized subcorpora as well as on the whole corpus. Results are summarized in Table 1 on the last page. Note that the brute force unigram baseline model, where all tokens that are present in the training corpus are assigned the most frequent tag from their ambiguity class, while unseen tokens receive the overall most frequent tag (singular nominative noun), performs remarkably well: this is due to the relatively large training corpus which results in a lower number of unseen items ($\approx 10\%$, compared to the 17.1% reported in Oravecz and Dienes (2002) who used a training corpus three times smaller).

As Table 1 makes clear, models such as WMA+T3 or 2Pass that have information on the tag sequence perform significantly better than models such as MA+ME that use only local information. The best combined model, 2Pass, outperforms all rule-based systems we know of that have been developed for Hungarian (Kuba et al., 2005; Horváth et al., 1999) under similar testing conditions and is more robust, though only slightly better, than the stochastic architecture of Oravecz and Dienes (2002). A clear advantage of our system over the others is its ability to robustly handle OOV items, making the processing of large heterogeneous corpora particularly efficient.

5. Conclusion and further work

In Hungarian, as in other highly inflecting languages, it is important to preserve detailed morphological information in the POS tags in order to provide useful clues for higher level processing tasks. This leads to a significantly larger tagset than is common in English (744 tags here as opposed

Subcorpus	Size	OOV	Unseen	Baseline	BMA	TnT	MA+ME	WMA+T3	2Pass
Literature	209785	5.79	12.19	86.20	95.46	96.02	97.63	97.63	97.83
Learner	290167	1.62	8.16	90.17	96.34	96.97	97.80	97.80	98.01
Press	355311	9.98	18.80	82.68	94.36	97.32	98.14	98.14	98.38
Technical	157969	8.43	15.04	86.06	94.44	97.02	97.91	97.91	98.11
Law	147766	4.97	8.13	91.41	96.89	98.44	98.96	98.96	99.04
Whole	1161016	5.64	9.59	89.70	95.40	97.42	97.93	97.93	98.17

Table 1: The performance of models

to the 36 standardly used in Treebank work), but does not degrade tagging performance, although it makes the training process computationally expensive.

In this paper we compared the performance of several POS tagging architectures developed for Hungarian. We have shown that stochastic components can be effectively combined with a symbolic morphological analyzer, and we have demonstrated that our best system reaches a performance level, 98.17%, that is comparable to state of the art English taggers. The resulting open source software system is remarkably robust in the face of OOV items, thereby allowing for efficient analysis of large heterogeneous (web-based) corpora.

Our future plans include a complete system that is entirely permissive in its license, without the current restrictions of TnT and SRILM. This system, currently in alpha, already achieves results comparable to those reported here.

6. References

Jason Baldridge, Thomas Morton, and Gann Bierner. 2001. The OpenNLP maximum entropy package. *http://maxent.sourceforge.net*.

Thorsten Brants. 2000. TnT – a statistical part-of-speech tagger. In *Proceedings of the Sixth Applied Natural Language Processing Conference (ANLP-2000)*, Seattle, WA.

Dóra Csendes, Jánós Csirik, and Tibor Gyimóthy. 2004. The Szeged Corpus: A POS tagged and syntactically annotated Hungarian natural language corpus. In Karel Pala Petr Sojka, Ivan Kopecek, editor, *Text, Speech and Dialogue: 7th International Conference, TSD*, pages 41–47.

Tomaž Erjavec, Sašo Džeroski, and Jakub Zavrel. 1999. Morphosyntactic tagging of Slovene: Evaluating pos taggers and tagsets. Technical Report 8018, Dept. of Intelligent Systems, Jožef Stefan Institute, Ljubljana.

Jan Hajič, Pavel Krbec, Karel Oliva, Pavel Květoň, and Vladimír Petkevič. 2001. Serial combination of rules and statistics: A case study in Czech tagging. In *Proceedings of the 39th Association of Computational Linguistics Conference*, pages 260–267, Toulouse, France.

Jan Hajič. 2000. Morphological tagging: Data vs. Dictionaries. In *Proceedings of ANLP-NAACL Conference*, pages 94–101, Seattle, Washington, USA.

Dilek Z. Hakkani-Tür, Kemal Oflazer, and Gökhan Tür. 2000. Statistical morphological disambiguation for agglutinative languages. In *Proceedings of the 18th conference on Computational linguistics*, pages 285–291, Saarbrücken, Germany.

Péter Halácsy, András Kornai, László Németh, András Rung, István Szakadát, and Viktor Trón. 2004. Creating open language resources for Hungarian. In *Proceedings of Language Resources and Evaluation Conference (LREC04)*. European Language Resources Association.

Tamás Horváth, Zoltán Alexin, Tibor Gyimóthy, and Stefan Wrobel. 1999. Application of different learning methods to Hungarian part-of-speech tagging. In *Proceedings of ILP*, pages 128–139.

András Kornai, Péter Halácsy, Viktor Nagy, Csaba Oravecz, Viktor Trón, and Dániel Varga. 2006. Web-based frequency dictionaries for medium density languages. In *Proceedings of the EACL 2006 Workshop on Web as a Corpus*.

András Kuba, László Felföldi, and András Kocsor. 2005. Pos tagger combinations on Hungarian text. In *2nd International Joint Conference on Natural Language Processing, IJCNLP*.

Andrew McCallum, Dayne Freitag, and Fernando Pereira. 2000. Maximum entropy Markov models for information extraction and segmentation. In *Proceedings of the 17th International Conference on Machine Learning*, pages 591–598. Morgan Kaufmann, San Francisco, CA.

Csaba Oravecz and Péter Dienes. 2002. Efficient stochastic part-of-speech tagging for Hungarian. In *Proceedings of the Third International Conference on Language Resources and Evaluation (LREC2002)*, pages 710–717.

Adwait Ratnaparkhi. 1996. A maximum entropy model for part-of-speech tagging. In Karel Pala Petr Sojka, Ivan Kopecek, editor, *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 133–142, University of Pennsylvania.

Noah A. Smith, David A. Smith, and Roy W. Tromble. 2005. Context-based morphological disambiguation with random fields. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, Vancouver.

Andreas Stolcke. 2002. SRILM – an extensible language modeling toolkit. In *Proceedings of the International Conference on Spoken Language Processing*.

Viktor Trón, György Gyepesi, Péter Halácsy, András Kornai, László Németh, and Dániel Varga. 2005. Hunmorph: open source word analysis. In *Proceeding of the ACL 2005 Workshop on Software*.

Viktor Trón, Péter Halácsy, Péter Rebrus, András Rung, Eszter Simon, and Péter Vajda. 2006. Morphdb.hu: Hungarian lexical database and morphological grammar. In *Proc LREC 2006*.