Transformation-based part-of-speech tagging for Serbian language

VLADO DELIĆ, MILAN SEČUJSKI, ALEKSANDAR KUPUSINAC
Faculty of Technical Sciences
University of Novi Sad
21000 Novi Sad, Trg Dositeja Obradovića 6
SERBIA
vdelic@uns.ac.rs, secujski@uns.ac.rs, sasak@uns.ac.rs http://www.ftn.uns.ac.rs

Abstract: – Machine learning techniques based on transformation rules have proven to be a viable alternative to stochastic tagging, achieving similar accuracy while having many advantages such as simplicity and better portability to other languages. However, data sparsity remains one of the greatest obstacles to tagging languages with complex morphology. Research in POS tagging for Serbian language described in this paper has resulted in several original ideas for improving tagging accuracy and overcoming problems related to data sparsity for highly inflected languages. The POS tagger for Serbian described in this paper achieves an error rate of 10.0% when trained on a previously annotated text corpus containing 190,000 words, which is comparable with results reported for some other languages with a similar level of inflection.

Key-Words: - natural language processing, POS tagging, transformation-based learning

1 Introduction

Practically every natural language processing system such as a speech synthesiser, machine translator or information retrieval system requires the input text to be processed in such a way that each word is assigned some specific additional information related to its morphological status, contained in a unique morphological descriptor or *tag*.

In case of languages with complex morphology, tags usually have specified internal structure, and their total number (*tagset size*) is much larger than in case of languages with simpler morphology. This, in turn, leads to the well-known problem of data sparsity, i.e. the fact that the amount of training data necessary increases rapidly with tagset size, making highly accurate POS taggers for such languages extremely hard to obtain.

The paper presents a transformation-based POS tagger developed for use within a speech synthesiser for Serbian language [1]. The accuracy of the basic procedure of transformation-based tagging has been improved by an efficient combination with tagging based on hidden Markov models, as well as two other language independent procedures especially useful for tagging texts in highly-inflective languages – inclusion of the optimum number of transformation rules for a specific pair of tags, and a novel procedure for generalisation of transformation rules.

2 The Basic Algorithm

Transformation-based part-of-speech tagging is an instance of the transformation-based learning (TBL) approach to machine learning, described in detail in [2]. The basic algorithm, introduced in [3], is based on sequential application of transformation rules obtained automatically, by analysis of a large sample of previously annotated text. As such, TBL tagger overcomes the common shortcomings of classical rule based approaches to natural language processing: it is robust and does not require (almost) any expert knowledge of grammar. Furthermore, it operates with a relatively small set of rules as opposed to a large amount of statistic data required by stochastic taggers to capture contextual information. Besides a significant reduction in stored information required, rules used by TBL taggers are easy to interpret by humans, unlike large tables of contextual probabilities. Such taggers are also easier to fine-tune manually, as well as portable to another tagset or even another language.

A TBL tagger is defined by its two key components:

- a specification of admissible types of error-correcting transformation rules,
- the learning algorithm.

The tagger requires a previously tagged corpus and a dictionary. Each word in the training corpus is initially assigned its most frequent tag, estimated by examining the training corpus without regard to context. The learning algorithm is then used to construct an ordered list of transformation rules that will be used to transform the initial tagging into one that is closer to correct. This list of rules will be used for tagging new text by again initially selecting the most frequent tag for each word, and then applying the transformations in a particular order.

Each of the transformation rules consists of a triggering environment and a rewrite rule. The triggering en-

vironment defines the conditions which have to be met for a rule to be considered fit for application. Rewrite rules, in the form $t_1 \rightarrow t_2$, define which source tag is to be replaced by which target tag. Triggering environments suggested in [3] are based on tags that appear up to three word positions left or right of the word whose tag is to be modified. An example of a transformation rule found by this tagger is:

TO IN NEXT-TAG AT,

stating that, if a word is tagged TO and the following word is tagged AT, then its tag should be switched to IN[1]. This rule is quite reasonable from a linguistic point of view, as is the case with most rules discovered in such a way. Once the list of rewrite rules has been acquired, new text can be tagged by initially assigning each word its most frequent tag regardless of context and subsequently applying transformation rules where possible. If the actual context of a word matches several different triggering environments, the rule to be used will be the one which resulted in the greatest error reduction when evaluated on the training corpus.

When tested on the Brown Corpus [4], containing about 1.1 million words from a variety of genres of written English, an error rate of 5.1% was obtained (90% of the corpus was used for the training of the initial lexical tagger, 5% were used for rule acquisition and another 5% for testing). However, it should be noted that the simple initial lexical tagger achieved an error rate of 7.9% to begin with, and that the application of transformation rules reduced the error by only 2.8%. Several strategies aimed at further improvement of the tagger, including its lexicalisation, were proposed in [2].

2.1 Tagging highly-inflective languages

Regardless of the actual tagging technique used, a number of modifications become necessary when dealing with highly inflective or agglutinative languages [5].

The most obvious dificulty is the larger number of words encountered in a sample text of the same size, when compared to languages such as English. For instance, it has been shown that an English corpus of about 250,000 words contains about 19,000 different words [6], whereas a Serbian corpus of the same size contains almost 46,000 different surface forms. The same problem has been reported for a wide variety of other languages (cf. e.g. [7]). Error rates are much higher if stochastic tagging procedures are applied directly to highly inflective or agglutinative languages. However, the problem related to handling unknown words can be alleviated by including a dictionary which essentially gives a better model of unknown words.

[1] TO = infinitive to, IN = preposition, AT = article, as defined in the Penn treebank tagset for English [3].

The other issue with such languages is a much larger amount of information contained in the morphology of a word. In languages with poor inflection a lot of information related to the syntactic function of a word is represented by word order or neighbouring function words. In highly inflective or agglutinative languages such information is marked on the word itself, and word order plays a minor role in marking syntactic function. This means that, for any NLP application requiring POS tagging as a preprocessing step, the tagger has to provide information related to all the relevant morphological categories (such as case or gender). In that way only will a POS tagger for a highly inflective or agglutinative language be as useful as a POS tagger without case or gender is for English. On the other hand, this leads to a significant increase in the size of the tagset, since tags consist of sequences of morphological tags rather than being unities [8]. With such enriched tagsets it is usually necessary to perform a morphological analysis of the unknown word form (data or dictionary based), and the task of a POS tagger then amounts to disambiguation among all the possible tags provided by the morphological analyser.

All corpus-based POS tagging techniques suffer from the well-known data sparsity problem. The training data available are far from sufficient for reliable estimation of statistical parameters, or alternatively, for identification of all pertinent transformation rules in case of TBL taggers. This problem is much more acute for highly inflective and agglutinative languages because of the size of the tagset. It is thus clear that if a tagging procedure evaluated on an English corpus is directly applied to such a language, dramatically inferior error rates can be expected [8], [9]. This is unsatisfactory from the point of view of practical application, yet some improvements are possible. The aim of the research described in this paper is to improve the performance of a TBL tagger for Serbian using several novel language-independent modifications.

3 Proposed modifications

3.1 Combining TBL with Markov models

One of the hypotheses examined in this research is that TBL tagging can be efficiently combined with stochastic tagging methods such as hidden Markov models (HMM) by introducing a HMM tagger as the initial tagger for TBL instead of assigning initial tags according to the relative frequency of tags in the training corpus. Both TBL and HMM taggers attempt to capture regularities in tag sequences and use them in tagging unknown text, but they do it in fundamentally different ways and have different drawbacks. HMM taggers lack flexibility, while

TBL taggers are very good at capturing complex tag patterns, but are unable to express the dependability of a rule in quantitative terms. Use of a TBL tagger for correcting the output of a HMM tagger exploits the advantages of both approaches, especially having in mind the general principle of TBL – that the accuracy of the initial system combined with a TBL system should never be lower than the original accuracy of the initial system. The HMM tagger used in the experiment was trained on the corpus used for rule acquisition.

3.2 Rule acquisition

The basic rule templates used by this algorithm are quite similar to those described in [2] and [3]. However, in the original version of the algorithm, transformation rules were selected using an iterative procedure including the evaluation of each candidate rule on a separate validation corpus within each iteration. While such an approach is feasible for languages with poor inflection, since e.g. for English an error rate of 5.1% can be achieved with as few as 71 transformation rules [3], for languages such as Serbian a more efficient procedure is desirable, since the number of relevant transformation rules can be expected to be much higher. For that reason, a rule acquisition procedure evaluating candidate rules in groups was adopted.

The first step of the procedure consists of the identification of all rules that reduce the error rate by any positive quantity. As was expected, the rules identified comprise only a small part of the entire search space defined by the tagset and the templates, since only a fraction of all possible instances of triggering environments actually appear in the corpus. The rules are organised into N groups according to their originating templates (N being 20 in the actual case), and a threshold value is adopted for each group, defining whether a transformation rule is to be included into the final list of error-correcting rules. Only the rules the application of which results in an improvement greater or equal to the threshold are considered as reliable and thus included into the list.

The decision to group the rules in this way and define thresholds depending on the originating template was motivated by the fact that some rule templates tend to instantiate a large number of rules that are generally unreliable, whereas some other templates instantiate a very small number of fairly reliable rules. An example of an unreliable rule template is:

> *"Change the tag from t_1 to t_2 in case any of the three following words is tagged t_3",*

whereas an example of a reliable rule template is:

> *"Change the tag from t_1 to t_2 in case the preceding word is tagged t_3 and the word before is tagged t_4".*

The threshold values are estimated within a fully automatic procedure similar to stochastic hill climbing. An initial value of all thresholds is adopted at random, defining an interim list of transformation rules. The list is evaluated on the validation corpus and the N-tuple defining thresholds is then modified by addition of a random N-tuple representing perturbation. The procedure is then repeated, eventually converging to a maximum, thus defining the final transformation rule list. The problem of arriving at local maxima significantly inferior to the global one can be partly alleviated by varying the standard deviation of the perturbation according to the recent history of error reduction values.

The method for rule acquisition described here includes a single evaluation of an entire cluster of rules per iteration rather than separate evaluations of individual rules, thus being significantly less time-consuming and suitable for use in tagging highly-inflective languages. The experiment has shown that the training process can sometimes take several hours on a standard PC configuration, depending on the initial threshold values selected at random, since each iteration requires that the entire corpus be tagged anew. Fortunately, the same does not hold for actual tagging, since once the final list of rules is established, its application is extremely fast, as is well known for transformation-based tagging.

3.3 Rule generalisation

Another problem that has been addressed within this research concerns the fact that, in case of inflectionally rich languages, specific transformation rules obtained by analysis of the corpus are usually instantiations of more general linguistic rules. On the other hand, each particular instance of a general rule usually cannot be found in the corpus due to data sparsity. For instance, the rule:

> *"Change the tag of an adjective to genitive plural feminine in case the following word is a genitive plural feminine noun",*

is an instance of a (hypothetical) general transformation rule:

> *"If a noun follows an adjective, change the values of the morphological categories case, number, and gender of the adjective to those assigned to the noun".*

A strategy able to infer general rules from a sample of their instances would be of great use for tagging highly inflected languages, since it would enable the tagger to perform correctly even in situations not explicitly present in the training corpus.

Before the introduction of a straightforward method for inference of general rules, it should be noted that a positional tag structure similar to those described in [10] and [11] is used. Each tag is thus a compact string representation of a simplified feature structure. The first

Source	Destination	Context	Instances	NUMBER				
				s	p	d	S	C
AAm**s**1-p-	AAm**s**4-p-	NNm**s**4---	174	174			174	174
AAm**p**2-p-	AAf**p**2-p-	NNf**p**2---	161		161		161	161
AAf**s**2-p-	AAf**p**1-p-	NNf**p**1---	128		128			128
AAm**s**1-p-	AAm**p**1-p-	NNm**p**1---	119		119			119
AAf**s**2-p-	AAm**p**4-p-	NNm**p**4---	112		112			112
AAm**s**2-p-	AAn**s**2-p-	NNn**s**2---	102	102			102	102
...				...				
AAm**p**1-p-	AAm**s**1-p-	NNm**s**1---	2		2		2	2
TOTAL:			1734	662	873		1056	1716

Table 1: Inference of the value of the morphological category *number* for $A \rightarrow A[N_{+1}]$ rules

character of each tag encodes only the major part-of-speech category (noun (N), verb (V), adjective (A), adverb (R), pronoun (P), preposition (S), numeral (M), conjunction (C), interjection (I), particle (Q), punctuation (Z) or "undefined" (X)). The second character encodes the "subpart of speech" and contains details about the major category. It can have 54 different values, all of them related to a particular value of the major category. For instance, verbs are divided into present (a), future (b), infinitive (c), etc. The following characters signify:

- *gender* (masculine (m), feminine(f) or neutral (n)),
- *number* (singular (s), plural (p) or dual (d)),
- *case* (values from 1 to 7 denoting appropriate cases as well as 7 letters denoting combinations thereof, e.g. (a) stands for "genitive or dative"[2]),
- *person* (values from 1 to 3),
- *degree of comparison* (positive (p), comparative (c) or superlative (s)),

whereas the remaining, eighth character is reserved for certain special uses. In case a certain morphological category is not applicable to a particular combination of features or a particular word, the value of that category is marked by a hyphen. Thus, for example, AAms1-p denotes the positive form of a nominative singular masculine adjective (adjectives not being marked for person or the feature reserved for special uses). The method for inference of general rules takes advantage from the fact that a standard positional tag system is used. Rules are once again considered in groups, this time classified according to the originating templates as well as part-of-speech values corresponding to the source tag, destination tag, as well as tags defining the triggering environment (context tags). For example, all rules stating that an adjective tag is to be replaced by an adjective tag with different values of morphological categories if followed by a noun tag ($A \rightarrow A[N_{+1}]$) are examined together, under the hypothesis that all these rules, or most of them, are instances of a general rule. In the corpus used within this research there were 27 specific rules of the type $A \rightarrow A[N_{+1}]$, some of them shown in Table 1, representing inference of the value of the morphological category *number* for the destination tag of the general rule.

Possible values for the morphological category *number* are *singular*, *plural*, and *dual*. The method for inference of general rules is based on the assumption that the value of the subtag representing *number* in the destination tag can be determined in one of the following ways:

- *assign singular (s),*
- *assign plural (p),*
- *assign dual (d),*
- *keep the original value from the source tag (S),*
- *copy the value from the context tag (C).*

The value of the morphological category *number* in the destination tag will be decided upon based on the method occurring most frequently in the training corpus. Table 1 shows that copying the value of the *number* category from the context tag covers 1716 of the 1734 instances of the $A \rightarrow A[N_{+1}]$ rule discovered in the training corpus. The same simple method applied to *case* and *gender* shows that the values of those categories should be copied from context tags as well.

It is clear that for a number of quadruples <*source tag, destination tag, template, context tag(s)*> there is no appropriate general rule. For that reason a general rule is accepted only if its application on the validation corpus results in greater error reduction than the application of specific rules only.

4 Experiment

The transformation-based POS tagger described in this paper was tested on the AlfaNum Text Corpus (ATC) containing approximately 200,000 words [12]. The corpus consists of sections from a variety of genres of writ-

[2] Such a value can be assigned e.g. to a preposition that precedes nouns in genitive or dative case.

ten Serbian language including newspaper articles, encyclopedic entries as well as fiction.

The task of a tagger is to select a single tag from a list of tags that may correspond to each surface form. This list can be provided by some kind of a morphological analyser or, alternatively, a dictionary. The tagger described in this paper relies on a dictionary containing about 100,000 lemmas (3.9 million inflected forms) and corresponding tags [13].

Out of 748 tags present in the dictionary, 703 of them actually appear in the corpus, 146 of them more than one hundred times. On the other hand, as much as 98.6% of the entire corpus is covered by the dictionary, with tag perplexity ranging from 1 (32.1% of ATC, punctuation marks excluded) up to 20 (a single entry in the entire ATC). The expected number of possible tags per surface form in the ATC is 2.93. It should, however, be kept in mind that this figure reflects the homonymy ratio of the Serbian language only with regard to the adopted tag structure, and is given here solely in order to set the baseline for this experiment at the accuracy of 45.7%.

In case no dictionary was used, some other strategy for tagging words not seen in the corpus would have to be adopted. However, the accuracy of strategies such as the one described in [3], attempting to deduce tags from word endings, tends to drop quickly with the increase in the tagset size, making them of little use in tagging morphologically rich languages. On the other hand, the accuracy of the strategy for tagging unseen words would have a great impact on the overall tagging accuracy, since a significant part of the test corpus consists of words not seen in the training corpus.

The dictionary is thus considered to be an indispensable resource for the task of initial tagging. The fraction of the test corpus not covered by the dictionary is initially marked X-------, signifying *undefined* part-of-speech category, and it is up to transformation rules to change that tag into the appropriate one based on context. No attempt was made at initial tagging of out-of-dictionary words using lexical rules, as suggested in [14], since the purpose of the dictionary was to be a single source for morphological analysis.

The introduction of a dictionary also had a very useful side-effect. Unlike the experiment described in [3], where the most of the available corpus had to be used for initial tagging and just a small fraction was left for acquisition of transformation rules, in the experiment described here the entire training corpus could be used for acquisition of transformation rules, which mitigated the effects of data sparsity to a certain extent.

The templates used in the experiment were the basic transformation rule templates suggested in [2] and [14]. Out of the total number of $N = 20$ templates used for the experiment, 11 make reference to tags and pairs of tags, while 9 make reference to words and pairs of words. No templates that include any kind of linguistic knowledge were used.

4.1 Results

In order to examine the influence of the size of the training corpus on tagging accuracy, a series of experiments was carried out, with training corpus size varying from 10,000 to 190,000 words, while the sizes of the validation and test corpora were kept constant at 10,000 words. Actual figures may vary slightly since cutting off corpora in mid-sentence was considered undesirable and therefore avoided. Since tagging of punctuation marks would be a trivial task, the results given here reflect only the accuracy of tagging orthographic words. The experiment examines both the usefulness of HMM (bigram model) as the initial tagger, as well as the efficiency of the rule generalisation procedure, as described in Section 3.3. Each round of the experiment was thus carried out four times: with/without using an HMM tagger for initial tag assignment, as well as with/without applying rule generalisation procedure. The results are shown together in Fig. 1 for the sake of comparison.

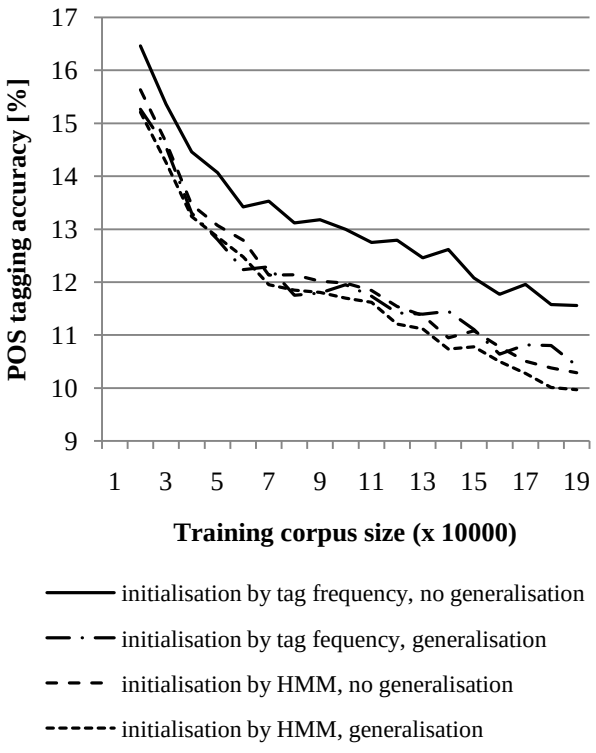

Fig. 1: POS tagging error vs. training corpus size

It can be seen that the results are comparable to the results reported in [10] for Czech and in [15] for Slovene, both Slavic languages with a similar level of inflection. However, it can be noted that rule generalisa-

tion, as described in Section 3.2, contributes to error reduction by at least 0.5% (the exact contribution depending on training set size).

The results confirm the well known fact that the level of accuracy of a POS tagger strongly depends on the complexity of morphological structure of the language. However, it should be kept in mind that POS tagging provides much more information in case of languages with a morphologically rich structure.

As to the total number of transformation rules identified and used in the experiment, it is indeed significantly greater than in the case of English. In case of frequency based initial tag assignment, the number of rules used reaches 11,392, while in case of HMM initial tag assignment it reaches 6,902, in both cases growing approximately linearly with training corpus size. On the other hand, the number of rules obtained by generalisation is smaller – regardless of the initial tag assignment method it reaches 60, growing at a lower rate and approaching saturation. A more detailed discussion on the numbers of transformation rules used is given in [13].

5 Conclusions and future work

The work described in this paper represents one of the first attempts at creating a completely automatic POS tagger for Serbian language. The algorithm includes a language independent modification of the existing transformational-based approach so as to make it more convenient for languages with morphologically rich structure, particularly highly inflected languages. This original variation on a simple theme of transformation-based tagging is especially useful when any reduction of the tagset size is unacceptable due to the requirements of the application.

Our future work will include investigation of more intelligent methods for rule generalisation as well as expanding the scope of the context. The last issue, which seems to be one of the greatest problems of POS tagging in general, is no less critical in case of languages with a high degree of inflection, since a clue for determining the correct value of a morphological category can often be found in a word or a set of words that are not in the immediate neighbourhood of the word being tagged.

References:

[1] M. Sečujski, V. Delić, D. Pekar, R. Obradović, and D. Knežević, An overview of the AlfaNum text-to-speech synthesis system, *Proc. of SPECOM*, Moscow, Russia, 2007. pp. 3-7 (Addenda Volume).

[2] E. Brill, Transformation-based error-driven learning and natural language processing: A case study in part-of-speech tagging, *Computational Linguistics*, Vol. 4, No. 21, 1995, pp. 543-566.

[3] E. Brill, A simple rule-based part of speech tagger, *Proc. of 3rd Int. Conf. on Applied Natural Language Processing, ACL*, Trento, Italy, 1992, pp. 152-155.

[4] W. N. Francis and H. Kučera, *Computational Analysis of Present-Day American English*, Brown University Press, Providence, RI, 1967.

[5] D. Jurafsky and J. H. Martin, *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*, Prentice-Hall, Upper Saddle River, NJ, 2000.

[6] C. Oravecz and P. Dienes, Efficient stochastic part-of-speech tagging for Hungarian, *Proc. of 3rd Int. Conf. on Language Resources and Evaluation, LREC*, Las Palmas, Canary Islands, Spain, 2002, pp. 710-717.

[7] D. Z. Hakkani-Tür, K. Oflazer, and Gökhan Tür, Statistical morphological disambiguation for agglutinative languages, *Journal of Computers and the Humanities*, Vol. 4, No. 36, 2002, pp. 381-410.

[8] J. Hajič, Morphological Tagging: Data vs. Dictionaries, *Proc. of 6th Applied Natural Language Processing and the 1st NAACL Conf.*, Seattle, WA, 2000, pp. 94-101.

[9] B. Hladká, *Czech Language Tagging (PhD Thesis)*. Institute of Formal and Applied Linguistics, Charles University, Prague, Czech Republic, 2000.

[10] J. Hajič and B. Hladká, Czech Language Processing – POS Tagging, *Proc. of 1st Int. Conf. on Language Resources and Evaluation, LREC*, Granada, Spain, 1998, pp. 931-936.

[11] C. Krstev, D. Vitas, and T. Erjavec, MULTEXT-East resources for Serbian, *Proc.of 8th Informational Society - Language Technologies Conf., IS-LTC*, Ljubljana, Slovenia, 2004, pp. 108-114.

[12] M. Sečujski, A software tool for automatic part of speech tagging in Serbian language, *Applied Linguistics*, Vol. 1, No. 9, 2008, pp. 97-103.

[13] M. Sečujski, *Automatic part-of-speech tagging of texts in the Serbian language (PhD Thesis)*, Faculty of Technical Sciences, University of Novi Sad, Serbia, 2009.

[14] E. Brill. Some advances in transformation-based part-of-speech tagging, *Proc. of 12th Nat. Conf. on Artificial Intelligence, AAAI*, Seattle, WA, USA, 1994, pp. 722-727.

[15] S. Džeroski, T. Erjavec, and J. Zavrel. Morphosyntactic Tagging of Slovene: Evaluating Taggers and Tagsets, *Proc. of 2nd Int. Conf. on Language Resources and Evaluation, LREC*, Athens, Greece, 2000, pp. 1099-1104.