The Odd Yearbook 8 (2010): 87–93, ISSN 2061-4896

A Hungarian NP Chunker

Gábor Recski and Dániel Varga

1 INTRODUCTION

In the following paper, we describe the preliminaries of a project aimed at creating an NP chunker for Hungarian with machine learning methods. First, we give a brief overview of the notion of chunks in natural language processing and describe the considerations behind the creation of the training data. Then we proceed to give a description of the chunker. Finally, we summarize the obtained results and give an outline of our further plans.

2 BACKGROUND

Abney (1994) describes chunks as discrete parts of a sentence which are relevant both for language comprehension (citing Gee & Grosjean 1983) and sentence prosody. He defines chunks as units that consist of ‘a single content word surrounded by a constellation of function words’ (Abney 1994: 1) and claims that it is the ordering of different chunks rather than their exact content which differs from language to language.

Abney reviews earlier definitions of chunks which called for a seperate chunk for each content word in a sentence and revises it to overcome some difficulties (e.g. those raised by embedded adjectives). He claims that each content word in a sentence is the rightmost word in a chunk, with the exception of content words between a function word and another content word which the function word selects (e.g. the adjective in the chunk *the proud man*). An example of the implementation of this definition is given by Abney and repeated in Figure 1. This definition overcomes difficulties such as that of a noun preceded by an adjective (which occurs in Hungarian as well), and yet it relies on a theoretical framework which makes use of the notion of syntactic selection (we shall soon see, however, that Abney is by no means the only author suggesting a definition of NP chunks with groundings in a procedural syntactic framework).

NP chunkers have been developed for several different languages, although most of them are for English. One of the most ground-breaking efforts was that of Ramshaw & Marcus (1995), who developed a learning algorithm which was trained on a data set derived algorithmically from a treebank and based primarily on part-of-speech (POS) tags of the target data; NP chunkers have followed these conventions ever since. The article

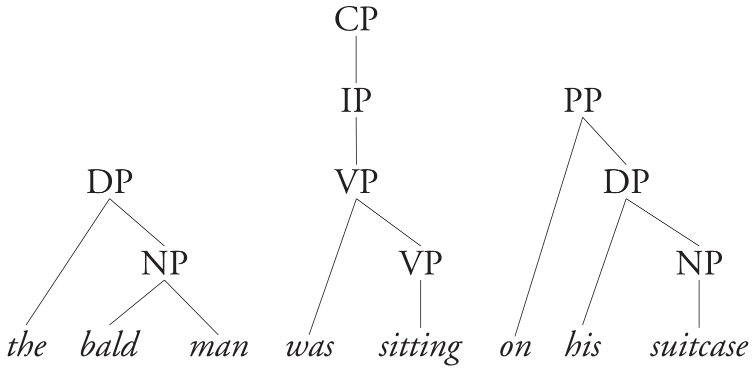

Figure 1: *Abney's chunks*

also reviews some previous approaches to the question of what to include in an NP chunk. Voutilainen (1993) introduces a method for identifying base NPs with the help of an extended set of POS-tags which automatically mark premodifiers of an NP as part of the chunk. Another approach is that of Bourigault (1992), who created French NP chunks in two phases: first generating what he called 'maximal length noun phrases' (ibid.: 980) and then extracting from them so-called *terminological units*. One of the earliest results in NP chunking is that of Church (1988) who inserts NP brackets into the POS-tagged *Brown Corpus*; however, he fails to provide details on how the training data was prepared, noting only that 'the training material was parsed into noun phrases by laborious semi-automatic methods' (ibid.: 141). Ramshaw and Marcus later reveal that Church's parser is incapable of handling several types of complex NPs, among them those that contain two coordinated noun phrases (Ramshaw & Marcus 1995). It would be a mistake, however, to compare results of the above works to each other or to those of our own since each of them refer to a slightly different and often inadequately documented task.

3 CREATING THE CORPUS

Since there has been no previous work on the chunking of Hungarian texts, our first task was to create a large set of training data. We therefore had to devise a method which would allow us to reduce a fully parsed corpus containing embedded phrases to one that is divided into discrete (i.e. non-overlapping) units. Taking the above theoretical considerations into account we were faced with the question of how to design our training data, that is, how to define Hungarian NP chunks for the first time. Our starting point was the *Szeged Treebank* (Csendes et al. 2004), a corpus created at the University of Szeged, which consists of 82,000 sentences with their complete syntactic structure. Since we expect our program to be able to identify all relevant noun phrases in a text, we decided to extract NP chunks by taking into account all NPs in the treebank which are not dominated by a higher level NP. Since this method yields chunks of various length and complexity, we included in the tagging a measure of complexity for each NP by assigning it a number that shows how many lower-level NPs it dominates. The chunking task does not involve identifying the level of an NP, but the presence of this information in the training corpus may aid the machine learning task.

4 SYSTEM ARCHITECTURE

4.1 Creating a labeling task

To solve the chunking task, we first turned it into a sequence labeling task. We marked each member of an NP with a tag that indicates whether it occupies the first (`B-N_x`), last (`E-N_x`) or any other position (`I-N_x`) within the chunk, or whether it constitutes an NP of its own (`1-N_x`). The `x` in `N_x` denotes the level of the NP.[1] Words outside of NPs were labeled `O`. Therefore the sentence analysed in the treebank as in Figure 2 will be labeled as in Table 1.

4.2 Feature extraction

Next, we proceeded to extract features from our corpus. The features of a word included its form, character trigrams and all pieces of morphological information available in the treebank. When tagging raw text, these latter features can be provided by the morphological disambiguator *hundisambig* (Halácsy et al. 2005), whose own errors, as we shall see, will only cause a slight decrease in performance.

4.3 The model

To model the labeling task, we used a Hidden Markov Model (HMM) (Rabiner 1989) with emission probabilities supplied by a Maximum Entropy

[1] By 'level of an NP' we mean a complexity measure: a maximal NP which does not dominate any lower-level NP received a complexity measure of 1, while every other chunk received the tag 2+ to indicate complexity of 2 or greater. This distinction was beneficial as it allowed for even finer distinctions to be made by the machine learning system. As there is no need for a tool to supply such complexity information about identified chunks in its output, this information is discarded at the end of the chunking process.

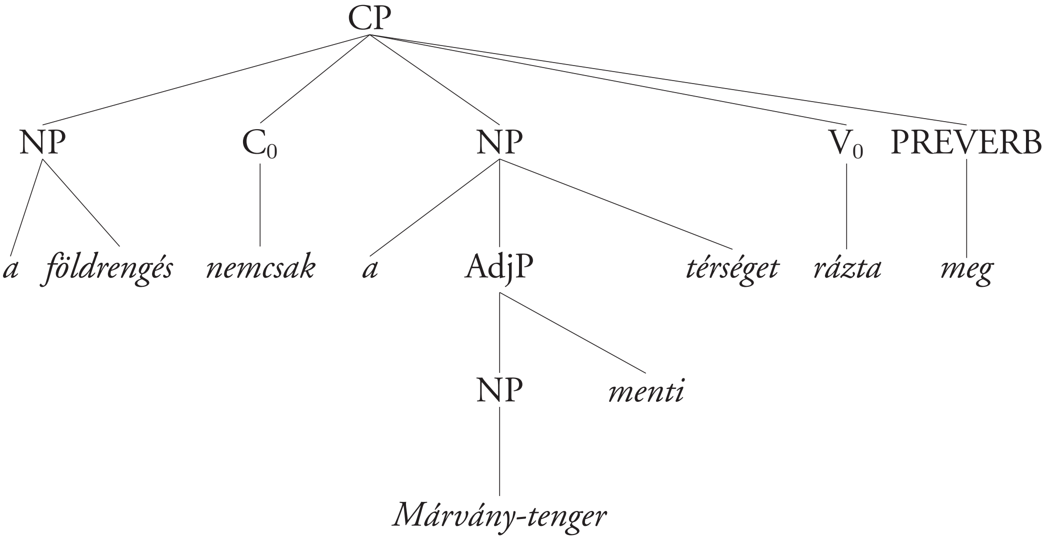

Figure 2: *Tree structure*

Word	Tag
A	`B-N_1`
földrengés	`E-N_1`
nemcsak	`O`
a	`B-N_2`
Márvány-tenger	`I-N_2`
menti	`I-N_2`
térséget	`E-N_2`
rázta	`O`
meg	`O`

Table 1: *Labeling*

model (Ratnaparkhi 1998). This has been shown to be a successful method in other supervised learning tasks for Hungarian, such as part-of-speech tagging (Halácsy et al. 2005) and named entity recognition (Varga & Simon 2006).

Let us now summarize the assumptions behind this model:

Let $p(i, u)$ denote the probability that the word in position i receives the tag u. We assume that the value of $p(i, u)$ depends solely on the features of the words in the context $w_{i-k} \dots w_{i+k}$. Hence $p(i, u)$ can be estimated by $\hat{p}(i, u)$ supplied by a maximum entropy model trained on these features.

Let $t(i, u, v)$ stand for the conditional probability that the word in position i receives tag u providing that the word in position $i - 1$ received the tag v. We assume that this probability is independent of i and estimate it by $\hat{t}(u, v)$, the conditional relative frequency directly observed in the training corpus.

During labeling, the system has to find the most likely tag sequence for a given sentence. If $\hat{p}(i, u)$ only depended on w_i (no context, just the current word), then the likelihood of a tag sequence could be written as a product thanks to conditional independence, and would be proportional to

$$\prod_i \frac{\hat{p}(i, u_i)\hat{t}(i, u_i, u_{i-1})}{P(u_i)}.$$

The maximum of this formula (that is, the best labeling) can be easily found by a Viterbi algorithm. This model is, in fact, the 'observations in states instead of transitions' version of maximum entropy Markov models, as suggested by McCallum et al. (2000). Our model can be described as a theoretically unfounded simple modification of this model: we let $\hat{p}(i, u)$ depend on a nontrivial $w_{i-k} \dots w_{i+k}$ $(k > 0)$ context rather than just w_i, and use the above formula as an approximation of the true likelihood. The optimum radius k of the context window was found to be 5 for these experiments.

5 EVALUATION

For the training task, we used a corpus of 1 million tokens; we tested the tagger on another 100,000 tokens. We evaluated the output along the guidelines of Sang & Buchholz (2000): *precision* and *recall* figures were calculated based on the output NPs and the actual set of NPs. The precision of a tagging is defined as the proportion of correctly tagged phrases to all tagged phrases. The recall is the proportion of correctly tagged phrases to all phrases in the corpus. Note that the chunker is trained on a corpus with information about the level of NPs. This means that the chunker can

	Precision	Recall	F-score
Baseline	60.24%	60.50%	60.37%
HunChunk	87.16%	84.99%	**86.06%**
HunDisambig + HunChunk	86.19%	84.20%	**85.18%**

Table 2: *Results*

provide such information. For the purposes of the evaluation, this information was discarded.

5.1 Baseline

Our baseline method was assigning the most probable tag to each word based on its part-of-speech tag. Using just two tags (`I-NP` for words within an NP and `O` for words outside of them), we reached a baseline F-score of only 51.03% (the F-score is the harmonic mean of the precision and recall of a system, used to represent the overall performance of the system). Tweaking the system only slightly, however (by introducing a third tag, `B-NP`, to mark words that are at the start of an NP) increased the F-score of the baseline system to 60.37%.

5.2 Results and conclusions

The obtained results are shown in Table 2. The last row shows the performance of the chunker when the morphological information is obtained from *hundisambig*, instead of the manually annotated Szeged Treebank.

In this paper we have described a system for identifying Hungarian noun phrases. We created an NP-corpus based on the Szeged Treebank and used it to train a Maximum Entropy model on the task of chunk-tagging, on the basis of which we created a statistical model for finding the most probable chunking for a given sentence.

At the time of this preliminary study, we are still experimenting with various learning parameters, different feature settings and with alternative machine learning algorithms. However, the above results seem to suggest that our system has the potential to become a useful component of a natural language processing toolchain.

REFERENCES

Abney, S. P. (1994). Parsing by chunks. Bell Communications Research.

Bourigault, D. (1992). Surface grammatical analysis for the extraction of terminological noun phrases. *Proceedings of the Fifteenth International Conference on Computational Linguistics*, pp. 977-981.

Church, K. W. (1988). A stochastic parts programs and noun phrase parser for unrestricted text. *Proceedings of ANLP-88*, Austin, TX.

Csendes, D., J. Csirik & T. Gyimóthy (2004). The Szeged Corpus: A POS Tagged and Syntactically Annotated Hungarian Natural Language Corpus. *Lecture Notes in Computer Science* 3206, pp. 41–47.

Gee, J. P. & F. Grosjean (1983). Performance structures: A psycholinguistic and linguistic appraisal. *Cognitive Psychology* 15, pp. 411–458.

Halácsy, P., A. Kornai & D. Varga (2005). Morfológiai egyértelműsítés maximum entrópia módszerrel. *Proceedings of the 3rd Hungarian Computational Linguistics Conference*, Szegedi Tudományegyetem.

McCallum, A., D. Freitag & F. Pereira (2000). Maximum Entropy Markov Models for information extraction and segmentation. *Proceedings of 17th International Conference on Machine Learning*, pp. 591-598.

Rabiner, R. L. (1989). A tutorial on Hidden Markov Models and selected applications in speech recognition. *Proceedings of IEEE 77:2*, pp. 257-286.

Ramshaw, L. A. & M. P. Marcus (1995). Text chunking using transformation-based learning. *Proceedings of the Third Workshop on Very Large Corpora*, Cambridge, MA.

Ratnaparkhi, A. (1998). *Maximum Entropy Models for Natural Language Ambiguity Resolution*. Ph.D. thesis, University of Pennsylvania.

Sang, E. F. T. K. & S. Buchholz (2000). Introduction to the CoNLL Shared Task: Chunking. *Proceedings of CoNLL 2000 and LLL 2000*, pp. 127-132.

Varga, D. & E. Simon (2006). Hungarian named entity recognition with a maximum entropy approach. *Acta Cybernetica* 16, pp. 293–301.

Voutilainen, A. (1993). NPtool, a Detector of English Noun Phrases. *Proceedings of Workshop on Very Large Corpora*, Ohio State University.