Brill’s Rule-Based Part of Speech Tagger
for Hungarian

Beáta Megyesi

Master’s Course in Computational Linguistics
D-level thesis, 10 credits
Advisor: Gunnel Källgren
Abstract

There are two main methodologies for automatic Part of Speech (PoS) tagging: statistical and rule-based. Almost all automatic PoS tagging systems are based on Markov models where training consists of learning lexical and contextual probabilities. Rule-based taggers on the other hand, work by rules that have been hitherto constructed by linguists. In spite of the fact that statistical models are less accurate than rule-based models most existing PoS analysers have been based on a probabilistic model because these systems are very robust and can be automatically trained. The limitations of the rule-based taggers are that they are non-automatic, costly and time-consuming.

Brill (1992) presented a rule-based PoS tagger which automatically infers rules from a training corpus based on transformation-based error-driven learning. Thereby Brill avoids most of the limitations of traditional rule-based taggers in his system.

In this study, Brill’s tagger is tested on Hungarian which, unlike English, has a very rich morphology, is agglutinative with some inflectional characteristics and has fairly free word order.

The tagging software has been trained on a Hungarian corpus, consisting of 99860 word tokens. This corpus has been annotated for part of speech including inflectional properties. The training was done twice, once for part of speech tags only and once for part of speech and ‘subtags’ denoting inflectional properties. New test texts with approximately 2500 words as well as the original training corpus were tested on the tagger. Precision was calculated for all test texts, and recall and precision for specific part of speech tags. The results presented in this work show that the accuracy of tagging of the test texts is 85.12% for part of speech tags alone, 82.45% for part of speech tags with correct and complete subtags, 84.44% for part of speech tags with correct but not necessarily complete subtags and 87.55% for part of speech tags without considering the correctness of the subtags.

The tagger has the most problems with parts of speech belonging to open classes because of their complicated morphological structure. On the other hand, grammatical categories are easier to detect and correctly annotate.

It is shown, that it is more difficult to derive information from words which are annotated with only PoS tags, than from words whose tags include information about the inflectional categories.

It is suggested that for better evaluation it would be necessary to test the tagger on a large corpus and also to evaluate different components of the system. Higher accuracy could probably be achieved by using a larger corpus made of different text types. Also, using a larger lexicon would reduce the number of unknown words. Additionally, accuracy could be greatly improved if the rule generating mechanisms in Brill’s system would be more flexible allowing consideration of the different characteristics of Hungarian and other languages. The results show, contrary to Brill’s statement that his system is language independent (1992), that Brill’s present system has some difficulty with languages which are dissimilar in their characteristics from English.
Acknowledgement

Above all, I would like to thank my advisor Gunnel Källgren for her suggestions and encouragement during the preparation of this work. I wish to express my deepest thanks to Klas Prütz for helping me with the training process of the tagger and for his many valuable suggestions. A very big thanks to Jennifer Spenader for reading the paper draft after draft, for her suggestions and comments, and for our many heated discussions concerning this work. I’m also grateful to Sofia Gustafsson-Capková for our fruitful meetings for discussing our work and to Christina Ericsson, Beb Lindberg and Sara Rydin for readings and comments. Thanks to Ljuba Veselinova for helping me with UNIX! I would also like to thank the researchers of the Research Institute for Linguistics, Hungarian Academy of Sciences, especially Júlia Pajzs and Csaba Oravecz for giving me the Hungarian corpora and to prof. Ferenc Kiefer for giving me the opportunity to participate in the project. Last but not least, I would like to thank my life-partner Olav Bandmann for his help and for his patience with me during this work.
Contents

1 INTRODUCTION ............................................................. 1

2 BACKGROUND ..................................................................... 3
  2.1 TYPES OF CORPORA ...................................................... 3
  2.2 PART OF SPEECH ANNOTATION ....................................... 4
  2.3 EVALUATION OF SYSTEM EFFICIENCY .......................... 6

3 BRILL’S RULE-BASED POS TAGGER ................................... 8
  3.1 TRANSFORMATION-BASED ERROR-DRIVEN LEARNING ....... 8
  3.2 LEARNING LEXICAL RULES ........................................ 10
  3.3 LEARNING CONTEXTUAL RULES ................................. 13
  3.4 TAGGING NEW TEXTS ................................................ 15
  3.5 SOME PRACTICAL INFORMATION ................................. 15

4 THE HUNGARIAN LANGUAGE .............................................. 16
  4.1 PHONOLOGY .............................................................. 16
  4.2 MORPHOLOGY ............................................................ 17
    4.2.1 Articles ............................................................. 17
    4.2.2 Nouns ............................................................... 17
    4.2.2.1 Number ......................................................... 18
    4.2.2.2 Person/Possession ......................................... 19
    4.2.2.3 Case ............................................................. 19
    4.2.3 Pronouns .......................................................... 20
    4.2.4 Adjectives ......................................................... 22
    4.2.5 Numerals .......................................................... 22
    4.2.6 Verbs ................................................................. 22
      4.2.6.1 Copula ......................................................... 24
      4.2.6.2 Infinitive ................................................... 24
      4.2.6.3 Participles .................................................. 24
      4.2.6.4 Verbal particles ......................................... 25
    4.2.7 Postpositions ...................................................... 25
    4.2.8 Adverbs ............................................................. 26
    4.2.9 Word formation .................................................. 26
      4.2.9.1 Word composition ....................................... 26
      4.2.9.2 Derivation of words .................................... 26
    4.3 SYNTAX ................................................................. 27
      4.3.1 Word order ....................................................... 27
      4.3.2 The main types of sentence structure ................... 28
      4.3.3 Agreement ....................................................... 29

5 THE HUNGARIAN CORPORA .............................................. 31

6 RESULTS AND EVALUATION OF SYSTEM EFFICIENCY ........ 36
  6.1 TESTING ON THE TRAINING SET ............................... 37
  6.2 TESTING ON THE POEM ............................................ 38
  6.3 TESTING ON THE FAIRY TALE ..................................... 40
  6.4 EVALUATION ............................................................ 42

7 DIRECTIONS FOR FUTURE RESEARCH ............................ 47

8 CONCLUSION ................................................................. 49

REFERENCES ................................................................. 50
APPENDIX A: THE HUNGARIAN TAGSET ............................................. 53
APPENDIX B: LEXICAL RULES ......................................................... 58
APPENDIX C: CONTEXTUAL RULES .................................................. 75
APPENDIX D: TEXT NORMALISATION ................................................. 92
APPENDIX E: OUTPUT ................................................................. 120
1 Introduction

There are two main methodologies for automatic Part of Speech (PoS) tagging: statistical (probabilistic) and rule-based. Almost all automatic PoS tagging systems are based on Markov models where training consists of learning lexical probabilities (P(Tag|Word)) and contextual probabilities (P(Tag_x|Tag_{x-1}^x), etc.). These taggers usually take a large manually annotated corpus from which they extract probabilities. Rule-based taggers on the other hand, work by rules that have been hitherto constructed by linguists.

In terms of accuracy, systems based on the statistical approach tend to reach 95-97% correct analysis using tagsets ranging from a few dozen to about 130 tags (see Charniak 1996; Church 1988; DeRose 1988; Garside et al. 1987; Merialdo 1994; etc.) while rule-based system, such as Constraint Grammar, can achieve slightly more than 99% correct results (Samuelsson & Voutilainen, 1997). In spite of the fact that statistical models are less accurate than rule-based models most existing PoS analysers have been based on a probabilistic model because of the robustness and automatic training ability of such systems. The limitations of the rule-based taggers are that they are non-automatic, costly and time-consuming.

In 1992 Eric Brill presented a rule-based system which differs from other rule-based systems because it automatically infers rules from a training corpus and it utilises some corpus statistics. It takes less space than probabilistic taggers. Hence, the system has become a competitive alternative to probabilistic tagging models. The tagger has been trained for tagging words for English with an accuracy of 97% correct (Brill, 1994a).

The aim of this study is to test Brill’s rule-based part of speech tagger on a Hungarian corpus. In this project, a Hungarian training corpus of 80 000 running words, George Orwell’s 1984, has been used. This corpus has been annotated for part of speech (PoS) including inflectional properties (subtags) by the Research Institute for Linguistics at the Hungarian Academy of Sciences. The tagger has been trained on the same material twice, once with PoS and subtags and once with only PoS tags. The question is, with what accuracy will the rules Brill’s tagger has inferred from the training corpus be able to tag new unannotated Hungarian texts. Hungarian, which belongs to the Finno-Ugric branch of the Uralic family, is different from most European languages. It has a very rich morphology, is agglutinative with some inflectional characteristics and has free word order. Thus, it is probable that the tagger trained for Hungarian will have a lower accuracy compared to the result for English. Furthermore, tagging systems of simple construction (for example Church and DeRose’s systems) are usually easier to develop and use for different languages while more sophisticated tagging systems (such as Brill’s system) in most cases are complicated to develop for languages of various types. The purpose of this study is to find out if Brill’s system is immediately applicable to a language, which greatly differ in structure from English, with a high degree of accuracy.

In section two, an overview of corpus types, part of speech annotation, different disambiguation strategies and methods of evaluating system efficiency are presented. Section three describes the theory and practice of the rule-based PoS tagger constructed by Eric Brill.

Section four presents a description of the Hungarian language. It contains a section about Hungarian phonology followed by a description of the morphology of the main parts of speech, such as nouns, pronouns, adjectives, verbs and different verbal particles, postpositions, and adverbs. After that, a short overview of word formation follows. The main characteristics of Hungarian syntax, such as sentence structure, word order and agreement are also described.

In section five, the Hungarian corpora which constitute the training corpus of the tagger and the test corpus with their two tagsets (PoS tags with and without inflectional information) are presented.
In section six, the training process of the tagger on the Hungarian corpus, the results of the tagging of different test texts and an evaluation of system efficiency of the tagger are presented and discussed. Precision and recall rates are shown for different parts of speech.

Section seven discusses ways in which the tagger could be developed to improve tagging of Hungarian and other grammatically similar languages as well as different limitations and advantages of the tagger.

This work has been carried out within the GRAMLEX COPERNICUS project (no: 621) which aims to forward research on building corpus linguistic tools (lemmatisers, concordance programmes, lexicons) for four languages: French, Italian, Polish and Hungarian. The Hungarian partners concentrate on the development of a morphological analyser and on making the result of the analysis more usable for lexicographic work. The main task of the Institute for Linguistics at the Hungarian Academy of Sciences in this project is to carry on research on homograph disambiguation for Hungarian texts. Testing and developing Brill’s rule-based system on Hungarian will hopefully give an effective tool for tagging new unannotated Hungarian texts.
2 Background

This section will present an outline of important notions concerning corpora, part of speech tagging, different methods for disambiguating words in a corpus and methods used to evaluate system efficiency.

2.1 Types of corpora

A corpus, from a computational linguist’s view, is a collection of machine-readable utterances, either spoken or written, taken as a representative sample of a given language, dialect or genre and used for linguistic analysis.

There are different ways in which corpora may be categorised, according to the type of texts and the types of annotation. According to Krenn & Samuelsson (1997) a corpus can be balanced, pyramidal or opportunistic. A balanced corpus consists of ‘different text genres of size proportional to the distribution of a certain text type within a language’ while pyramidal corpora ‘range from very large samples of a few representative genres to small samples of a wide variety of genres’ (Krenn & Samuelsson, 1997:64). Opportunistic corpora consist of those texts which the collector can get.

A corpus can be classified as unannotated or annotated. Unannotated corpora contain raw but in most cases pre-processed texts, i.e. tokenised text with control characters eliminated. (More about tokenisation follows in section 2.2.)

Corpus annotation involves a repository of several types of linguistic information. The input to an annotated corpus is a raw text and the output is an analysed, annotated text. Labels or tags are attached to segments of the text, describing their properties or relations between them. The information about the text may include the origin and the title of the text, the author’s name, the declaration of the writing system, etc. Linguistic information includes part of speech annotation, lemmatisation, parsing, semantics, discourse and text linguistic annotation, phonetic transcription and prosody (McEnery & Wilson, 1996).

A tag may be defined as a code representing some feature or a set of features and is attached to the segments in a text. The tag can convey single or complex information. According to McEnery & Wilson (1996) there are several requirements on the characteristics of a tag. One is the basic idea that there should be only one part of speech tag per unit. Another claim is the divisibility of the tag, e.g. each element in a tag should mean something in itself. The third is mnemonicity - the ability to see in an easy way from a tag its meaning.

The tagsets may represent morphological, syntactic and/or phrase structure information. It can incorporate some special tags for foreign words, unknown words, idioms, symbols, etc. Thus, the size, contents, complexity and level of annotation of a tagset varies from system to system. How the tagset categorises words and phrases is also different from system to system.

Tagging can be done manually or automatically. A manually tagged corpus is annotated by a person or a group of people while an automatically annotated corpus is done by one or more computer programs whose output may or may not be corrected by human beings. Manual annotation of texts has several disadvantages. It is costly, and time-consuming. Often the work is done by a fairly large group of people which results in the additional problem of inconsistency of annotation that is often difficult to detect and correct. With automatic tagging the number of errors may be larger than with manual tagging, but the errors are consistent and are easier to detect. The drawback with automatic tagging is the problem of disambiguation which greatly increases the number of errors in tagging. It is of great importance that every annotation (or tag) is clearly defined from the start to avoid inconsistency in tagging, whether it is done manually or automatically.

Corpora can be used, for example, for natural language statistics, machine translation, natural language processing, speech recognition, etc. Corpora are also used for training,
testing and evaluation methods for disambiguation strategies. Large linguistically annotated corpora are used for training speech processing and stochastic part of speech taggers and parsers (Krenn & Samuelsson, 1997).

### 2.2 Part of speech annotation

The most basic type of corpus annotation is part of speech tagging (PoS-tagging), which is necessary in most applications of natural language processing (NLP), speech synthesis, speech recognition, machine translation, etc. PoS-tagging is useful for further linguistic study, for analysing the syntactic structure of the sentences in a text or for statistical work such as counting the distribution of the different word classes in text corpora.

PoS-tagging consists of assigning a word the accurate part of speech tag, i.e. the text is annotated with syntactic category at word level, but also other relevant information such as the inflectional categories of the classes, e.g. number, person and case. Thus, the task of PoS-tagging is attaching appropriate grammatical or morphosyntactical category labels to every word token, and even to punctuation marks, symbols, abbreviations, etc. in a corpus.

Words traditionally have been divided into different word classes on the basis of their meaning, form, or syntactic function, (e.g. in English, nouns, pronouns, verbs, adverbs, adjectives, prepositions, conjunctions, and interjections are generally recognised). The great problem is deciding what criteria (semantic, morphological and/or syntactical) should be used for classifying words into a certain part of speech. The size of the tagset varies from system to system depending on how detailed the categorisation of words is. For instance, one tagset e.g. the LOB-tagset\(^1\), can make a difference between lexical verbs, modal verbs and irregular verbs, while other systems such as the Penn-tagset\(^2\) uses the same tag (VB) for all verbs and distinguishes only the morphosyntactic forms of the verb (McEnery & Wilson, 1996).

Another major problem in automatic PoS-tagging is that words may belong to several parts of speech, i.e. are ambiguous between two or more classes, e.g. the word *graduate* belongs to either the class verb, noun or adjective. Contextual (and perhaps prosodic) information is needed to decide to which class the item belongs. Part of speech annotation has to handle the disambiguation of homographs, known and unknown words as well as achieve a high rate of correctly tagged words.

A PoS-tagger usually consists of three components: a tokeniser, a morphological classifier and a morphological disambiguator. The tokeniser segments the input text into words and sentences and (sometimes) recognises proper names, acronyms, and idioms by means of dictionaries. After tokenisation, the morphological classifier identifies all the possible word-classes and morpho-syntactic features (e.g. number, gender, case, etc.) of each word according to a lexicon. For non-identified words, the morphological analyser is used to detect if the lexeme exists in the lexicon or not by guessing common word endings. The larger the lexicon, the greater the chances will be that the words can be identified and assigned the appropriate part of speech tag. Then, the morphological disambiguator tries to disambiguate each word in a corpus by making the correct decision about the appropriate part of speech tag for a word in a certain context. Thus, automatic part of speech taggers take raw text, one or several lexicons or lists (e.g. for idioms), a morphological analyser and some rules or transition probabilities for disambiguation and produce a part of speech description of the text.

---

1. The LOB-tagset is designed for tagging the LOB corpus (Lancaster/Oslo-Bergen corpus) which is a collection of more than 1 000 000 words of English text taken from different sources (de Marcken, 1990).
2. The Penn-tagset is designed for the Penn treebank which is a PoS tagged and parsed corpus consisting mainly of the Brown Corpus and articles from the Wall Street Journal (Marcus, Marcinkiewitz, Santorini, 1993).
Identifying all possible part of speech tags of a word is regarded as ‘an easy problem’ while disambiguation is known to be a difficult problem. Mainly, there are two types of disambiguation strategies which are frequently used in systems of today: statistical/probabilistic approaches and rule-based methods.

The main goal of the probabilistic methods, given a sequence of words $W_1...W_n$ is to find the sequence of tags $T_1...T_n$ that is the most likely tag sequence for these words, i.e. the sequence $T_1...T_n$ that maximises $P(T_1...T_n | W_1...W_n)$.

It is possible to estimate $P(T_1...T_n | W_1...W_n)$ using the probabilities $P(T_i | W_i)$ and $P(T_i | T_{i-1}T_{i-2})$ (see Krenn and Samuelsson, 1997:97). These two probabilities are called lexical and contextual probabilities, respectively. Lexical probability is the probability of observing a tag given a specific word $P(T_i | W_i)$. Contextual probability, on the other hand, is the probability of observing a specific tag ($T_i$) given the surrounding tags. They are typically implemented as bigrams or trigrams, i.e. $P(T_i | T_{i-1})$ or $P(T_i | T_{i-1}T_{i-2})$, respectively. The lexical and contextual probabilities (frequency-based information) are often estimated from a correctly annotated (pretagged) training corpus.

Thus, various specific disambiguation algorithms which have been developed for part of speech taggers based on probabilistic methods, decide upon one, the most probable of the several possible categories for a given word, i.e. they disambiguate categorical ambiguities in a given text based on statistical methods (see Church 1988; Cutting et al 1992; DeRose 1988; Garside 1987; Merialdo 1994; Weischedel et al. 1993, etc.). One example of a system based on the probabilistic approach is CLAWS (Constituent Likelihood Automatic Word-Tagging System), which is a PoS tagger for English trained on a manually analysed text (see Garside 1987; Atwell 1993; Prütz 1996; CLAWS, 1998). CLAWS consists of three stages: pre-edit, automatic tag assignment, and manual post-edit. In the pre-edit stage the machine readable text is automatically converted to a suitable format for the tagging program. The text is then passed through the tagging program which assigns a PoS tag to each word or word combination in the text. CLAWS has a lexicon of words with their possible parts of speech, and a list of lexicalised phrases or idioms. For unknown words which are not in the databases, CLAWS uses various heuristics including a suffix list of common word suffixes with their possible parts of speech. The disambiguation component of the system selects the most likely tag for each word by calculating the probability for all possible paths through the sequence of ambiguously tagged words on the basis of a matrix of transitional probabilities derived from a training corpus. Finally, manual post-editing using a special tag editor may take place to correct the machine output. The CLAWS system has an accuracy of between 96%-97%, depending on the type of text.

DeRose (1988) created another system, the VOLSUNGA algorithm, which is similar to CLAWS and operates in linear rather than exponential time and space. The VOLSUNGA algorithm is based on Dynamic Programming meaning that a ‘mean problem’ can be divided into subproblems and that the main problem can be solved faster if the best solution of the subproblems are available. DeRose’s system has an accuracy of 96%.

In contrast to the probabilistic method, the rule-based approach for disambiguation uses rules created by linguists or by a computer program. The rules may contain a large number of morphological, lexical and/or syntactical information. They are based on linguistic knowledge about the structure of the language and specific languages. With human rule creation there is a large set of manually constructed rules based on a specific grammar, written in a formal notation (see example a below) so that they can be used by the computer for further parsing. Many systems also use the trial-and-error method, i.e. finding sentences where rules have failed in order to manually add further rules to the system for higher accuracy. The drawback of this method is that adding a rule to the system may involve overgeneration, i.e. one extra rule can result in more harm than good to the accuracy of general tagging. Rule-based methods are time-consuming and require a great knowledge of both languages in general and the specific language being tagged.

TAGGIT, constructed by Greene and Rubin (1971) was one of the first systems which used rules for disambiguation with an accuracy of 77%.
Another example of a rule-based system is Constraint Grammar (CG), which is a very successful language independent rule-based approach. It has an accuracy of 99.7%, including some ambiguities. The tagger uses a two-level morphological analyser with a big lexicon and a large number of morphological and syntactical descriptions, created by linguists. Rules are accessible for several languages, such as Finnish, English and Swedish (see Samuelsson & Voutilainen, 1997).

A rule from CG may look like the one presented in example a. This rule means ‘discard a verb interpretation (first line) if the word is preceded by an unambiguous determiner (second line)’.

Example A

@w = 0 (V)
(-1C DET).

CG, like many other models, is intended to be language independent when the necessary language specific information is also implemented.

Unlike TAGGIT or CG, Brill’s rule-based tagger automatically creates rules from a training corpus, without human intervention. Brill’s tagger will be described in more detail in section 3, but first some explanation about the evaluation of system efficiency.

2.3 Evaluation of system efficiency

System efficiency may be evaluated by two metrics for effectiveness: recall and precision. Recall is the ratio of the number of retrieved intended tags returned by the tagger divided by the total number of intended tags. Precision is the ratio of the number of all retrieved intended tags, divided by the total number of retrieved tags.

Consider figure a below for a detailed explanation.

Figure A. The representations of the different sets, where A is the set of intended items, B is the set of retrieved items, x, y and z are the number of elements corresponding to the different parts in the picture.

Let A be the set of intended items, let B be the set of retrieved items. x, y, and z are the number of elements corresponding to the different parts in figure a above. Recall and precision are defined as following:

\[
\text{recall} = \frac{y}{y+x} = \frac{\text{intended}_\text{found}}{\text{intended}_\text{total}}
\]

\[
\text{precision} = \frac{y}{y+z} = \frac{\text{intended}_\text{found}}{\text{total}_\text{found}}
\]

Recall estimates the probability that the tagger annotates a word with tag T given that the correct annotation for the word is T. Precision estimates the probability that the correct annotation for a word is T given that the tagger annotates a word with T. For example, a recall of 90% means that the system missed 10% of the intended tags while a precision of 85% means that 15% of what the system retrieved was not regarded as a correct result.
The error rate is defined as

\[
\text{error rate} = 100\% - \text{precision}.
\]

The effectiveness of PoS taggers is measured in terms of accuracy, i.e. the correctly assigned tags in a text. The perfect system could be characterised as 100% recall and 100% precision. The accuracy of most systems of today is between 95% and 99%. According to Krenn & Samuelsson (1997) the accuracy depends on the following conditions with examples: i) the size of the tagset, for example, a small tagset results in good results when trained on a small corpus, ii) the size of the training corpus, e.g. the bigger the tagset the less correct the result will be if the size of the training corpus doesn’t increase accordingly, iii) the type of training and test corpora, e.g. if a tagger is trained on a specific training corpus and is tested on another type of text, the result will be less satisfactory than with a similar test corpus and lastly, iv) the type and the completeness of the vocabulary, e.g. an incomplete lexicon also results in less accuracy.

The theoretical background for evaluation of system efficiency comes from information retrieval research. More information and a nice overview about this topic is found in Krenn & Samuelsson (1997).
3 Brill’s rule-based PoS tagger

Eric Brill introduced a PoS tagger in 1992 that was based on rules, or transformations as he calls them, where the grammar is induced directly from the training corpus without human intervention or expert knowledge. The only additional component necessary is a small, manually and correctly annotated corpus - the training corpus - which serves as input to the tagger. The system is then able to derive lexical/morphological and contextual information from the training corpus and ‘learns’ how to deduce the most likely part of speech tag for a word. Once the training is completed, the tagger can be used to annotate new, unannotated corpora based on the tagset of the training corpus.

The rule-based part of speech tagger can be said to be a hybrid approach, because it first uses statistical techniques to extract information from the training corpus and then uses a program to automatically learn rules which reduces the faults that would be introduced by statistical mistakes (Brill, 1992). The tagger does not use hand-crafted rules or prespecified language information, nor does the tagger use external lexicons or lists of different types. According to Brill (1992) ‘there is a very small amount of general linguistic knowledge built into the system, but no language-specific knowledge’.

The long time goal of the tagger is, in Brill’s own words, to ‘create a system which would enable somebody to take a large text in a language he does not know and with only a few hours of help from a speaker of the language accurately annotate the text with part of speech information.’ (Brill & Marcus, 1992b:1). For achieving his aim, Brill has also developed a parser, consisting of systems which may automatically derive word classes and the bracketing structure of sentences, assigning nonterminal labels to the bracketing structure and improving prepositional phrase attachment (see Brill, 1992, 1993a, 1993b, 1994a, 1995a, 1995b).

In this work the part of speech tagger, applying the most likely tag for a word will be presented. The following section gives a description of the main ideas of Brill’s tagger and some information about how to train and test that tagger. All information contained in the following section is based on Brill’s articles, listed in the reference list (Brill, 1992; 1993a; 1993b; 1993c; 1994a; 1995a; 1995b). The description below is only functional, disregarding any efficiency aspects considered in the actual software.

3.1 Transformation-based error-driven learning

The general framework of Brill’s corpus-based learning is so-called Transformation-based Error-driven Learning (TEL). The name reflects the fact that the tagger is based on transformations or rules, and learns by detecting errors. First, a general description of how TEL works in principle is given, then a more detailed explanation for the specific modules of the tagger will follow in subsequent sections.

Roughly, the TEL, shown in figure b, begins with an unannotated text as input which passes through the initial state annotator. It assigns tags to the input in some fashion. The output of the initial state annotator is a temporary corpus which is then compared to a goal corpus which has been manually tagged. For each time the temporary corpus is passed through the learner, the learner produces one new rule, the single rule that improves the annotation the most (compared with the goal corpus), and replaces the temporary corpus with the analysis that results when this rule is applied to it. By this process the learner produces an ordered list of rules.

The tagger uses TEL twice: once in the lexical module deriving rules for tagging unknown words, and once in the contextual module for deriving rules that improve the accuracy. Both modules use two types of corpora: the goal corpus, derived from a manually annotated corpus, and a temporary corpus whose tags are improved step by step to resemble the goal corpus more and more.
In the lexical module of the tagger the goal corpus is a list of words containing information about the frequencies of tags in a manually annotated corpus. A temporary corpus on the other hand is a list consisting of the same words as in the goal corpus, tagged in some fashion. In the contextual learning module the goal corpus is a manually annotated running text while a temporary corpus consists of the same running text as in the goal corpus but with different tags.

A rule consists of two parts: a condition (the trigger and possibly a current tag), and a resulting tag. The rules are instantiated from a set of predefined transformation templates. They contain uninstantiated variables and are of the form

\[
\text{If Trigger, then change the tag } X \text{ to the tag } Y \\
\text{or} \\
\text{If Trigger, then change the tag to the tag } Y
\]

where X and Y are variables. The interpretation of the first type of the transformation template is that if the rule triggers on a word with current tag X then the rule replaces current tag with resulting tag Y. The second one means that if the rule triggers on a word (regardless of the current tag) then the rule tags this word with resulting tag Y. The set of all permissible rules PR are generated from all possible instantiations of all predefined templates.

The rule generating process takes an initially tagged\(^3\) temporary corpus TC\(_0\) and finds the rule in PR which gets the best score when applied to TC\(_0\) (a high score for a rule means that the temporary corpus produced when applying the rule gives an annotation closer to the goal corpus). Call this rule R1. Then R1 is applied to TC\(_0\), producing TC\(_1\). The process is now repeated with TC\(_1\), i.e. it finds the rule in PR which gets the best

---

\(^3\) Initial state annotation is done differently depending on which learning module is used. This will be explained in detail later.
score when applied to $TC_1$. This will be rule R2\(^4\) which then is applied to $TC_1$ producing $TC_2$. The process is reiterated producing rules R3, R4, ... and corresponding temporary corpora $TC_3$, $TC_4$, ... until the score of the best rule fails to reach above some predetermined threshold value. The sequence of temporary corpora can be thought of as successive improvements closer and closer to the goal corpus. Note that only the tags differ in the different temporary corpora. The output of the process is the ordered list of rules R1, R2, ... which will serve to tag new unannotated texts.

The score for a rule $R$ in PR is computed as follows. For each tagged word in $TC_i$ the rule $R$ gets a score for that word by comparing the change from the current tag to the resulting tag with the corresponding tag of the word in the goal corpus. How this is computed depends on the module, see sections 3.2 and 3.3 below. A positive score means that the rule improves the tagging of this word, and a negative score means that the rule worsens the tagging. If the condition of the rule is not satisfied then the score is zero. The total score for $R$, score($R$), is then obtained by adding the scores for each word in $TC_i$ for that rule. When the total score for each $R$ is obtained the rule which has the highest score is added to the set of rules which have already been learned. Rules are ordered, i.e. the last rule is dependent on the outcome of earlier rules.

The following sections present an overview of the two modules in Brill’s system: the lexical module and the contextual module. First, the lexical module uses transformation-based error-driven learning to produce lexical tagging rules. Then the contextual module, also using transformation-based error-driven learning, produces contextual tagging rules. The lexical and the contextual modules have different transformation templates. The triggers in the lexical module depend on the character(s) and the affixes of a word and some of them depend on the following/preceding word. The triggers in the contextual module, on the other hand, depend on the current word itself, the tags or the words in the context of the current word.

### 3.2 Learning lexical rules

The ideal goal of the lexical module is to find rules that can produce the most likely tag for any word in the given language, i.e. the most frequent tag for the word in question considering all texts in that language. The problem is to determine the most likely tags for unknown words, given the most likely tag for each word in a comparatively small set of words.

The lexical learner module is weakly statistical. It uses the first half of the manually tagged corpus\(^5\) as well as any additional large unannotated corpus containing the manually tagged corpus with the tags removed. The learner module uses three different lexicons or lists constructed (by the user) from the manually tagged corpus and the large unannotated corpus. The smallwordtaglist, built from the manually annotated corpus, serves as the goal corpus in TEL and contains lines of the form

\[
\text{Word Tag Frequency.}
\]

It simply contains the frequencies of each Word/Tag pair in the manually annotated corpus. Freq($W,T$) denotes the Frequency of a word $W$ with a specific tag $T$, and Freq($W$) denotes the frequency of the word $W$ in the manually annotated corpus. These numbers are used by the lexical learner in two ways: Freq($W,T$) is used to compute the most likely tag $T$ for the word $W$ (i.e. the one with the highest frequency), and

\[^4\] Note that R2 could in principle be the same rule as R1, i.e. the rules considered in each iteration is the whole set PR.

\[^5\] The second half of the manually annotated corpus will be used with the contextual rule learner.
\[
P(T|W) := \frac{\text{Freq}(W,T)}{\text{Freq}(W)}
\]
is the estimated probability that the word \(W\) is tagged with the tag \(T\).

The other two lists are called bigwordlist and bigbigramlist and are constructed from the large unannotated corpus. bigwordlist is a list of all words occurring in the unannotated corpus, sorted by decreasing frequency. The bigbigramlist is a list consisting of all word pairs (hence the name bigram) occurring in the unannotated corpus. Bigbigramlist does not contain the frequencies of word pairs; it only records if a given word pair occurs in the unannotated corpus or not. Note that once the user has constructed these lists, the lexical learner does not need either the manually annotated or the unannotated corpus because the lists contain all the information the learner needs. The bigwordlist and the bigbigramlist are only used to check the trigger condition, which is completely determined by the data in these two lists.

First, the learner constructs a word list from smallwordtaglist, i.e. the words with tagging information removed. Then the initial state annotator assigns to every word the default most likely tag. The default tags for English are NN and NNP\(^6\), where NNP is assigned to words which start with a capital letter and NN is assigned to words which do not start with a capital letter. The word list thus obtained is the initial temporary corpus \(WL_0\); it was called TC\(_0\) in the general description of TEL above.

Once \(WL_0\) has been produced by the initial state annotator the learner generates the set of all permissible rules \(PR\) from all possible instantiations of all the predefined lexical templates and computes a score for every rule \(R\) in \(PR\) (see below). The rule which achieves the best score becomes rule number one in the output. Then the learner transforms \(WL_0\) to \(WL_1\) by applying this rule. This process is repeated until no rule can be found with a score greater than some threshold value, i.e. compute the new scores for all the rules in \(PR\), pick the one with the best score, output this rule as rule number two and apply it on \(WL_1\) to get \(WL_2\), etc.

The scoring function is defined as follows: If the rule \(R\) has the template:

\[
\text{if Trigger then change tag } X \text{ to tag } Y
\]

and \(w\) is a word in \(WL_i\) with current tag \(X\) satisfying the trigger condition, then \(R\) gets the score \(P(Y|w) - P(X|w)\) for the word \(w\). The total score for \(R\) is then obtained by adding all the ‘word scores’.

\[
\text{score}(R) := \sum_{w} P(Y|w) - P(X|w)
\]

where the sum runs through all \(w\) in \(WL_i\) with current tag \(X\), satisfying the trigger condition.

If the rule \(R\) has the template:

\[
\text{if Trigger then change current tag to tag } Y
\]

and \(w\) is a word in \(WL_i\) satisfying the trigger condition, then \(R\) gets the score \(P(Y|w) - P(\text{Current tag of } w|w)\) for the word \(w\). The total score for \(R\) is then obtained by adding all the ‘word scores’.

\[
\text{score}(R) := \sum_{w} P(Y|w) - P(\text{Current tag of } w|w)
\]

\(^6\) In the case of the Hungarian corpus I simply used these default tags as nonterminal tags.
where the sum runs through all $w$ in $WL$, satisfying the trigger condition. Note that the score the rule $R$ gets for $w$ always is of the form $P(\text{new tag}|w)-P(\text{old tag}|w)$. A positive score means that the new tag is more likely than the old tag while a negative score means that the new tag is less likely than the old tag. The trigger condition is tested using $\text{bigwordlist}$ and $\text{bigbigramlist}$, and the estimated probabilities are computed from the frequencies in $\text{smallwordtaglist}$.

The set of templates used and the name of the rule type for each template in parenthesis are given below, (see also Appendix B).

1. Change the most likely tag to $Y$ if the character $Z$ appears anywhere in the word. $(\text{char})$
2. Change the most likely tag to $Y$ if the current word has suffix $x$, $|x| \leq 4$. $(\text{hassuf})$
3. Change the most likely tag to $Y$ if deleting the suffix $x$, $|x| \leq 4$, results in a word. $(\text{deletesuf})$
4. Change the most likely tag to $Y$ if adding the suffix $x$, $|x| \leq 4$, results in a word. $(\text{addsuf})$
5. Change the most likely tag to $Y$ if the current word has prefix $x$, $|x| \leq 4$. $(\text{haspref})$
6. Change the most likely tag to $Y$ if deleting the prefix $x$, $|x| \leq 4$, results in a word. $(\text{deletepref})$
7. Change the most likely tag to $Y$ if adding the prefix $x$, $|x| \leq 4$, results in a word. $(\text{addpref})$
8. Change the most likely tag to $Y$ if word $W$ ever appears immediately to the left/right of the word$^8$. $(\text{goodleft})$ vs. $(\text{goodright})$
9. Change the most likely tag from $X$ to $Y$ if the character $Z$ appears anywhere in the word. $(\text{fchar})$
10. Change the most likely tag from $X$ to $Y$ if the current word has suffix $x$, $|x| \leq 4$. $(\text{fhassuf})$
11. Change the most likely tag from $X$ to $Y$ if deleting the suffix $x$, $|x| \leq 4$, results in a word. $(\text{fdeletesuf})$
12. Change the most likely tag from $X$ to $Y$ if adding the suffix $x$, $|x| \leq 4$, results in a word. $(\text{faddsuf})$
13. Change the most likely tag from $X$ to $Y$ if the current word has prefix $x$, $|x| \leq 4$. $(\text{fhaspref})$
14. Change the most likely tag from $X$ to $Y$ if deleting the prefix $x$, $|x| \leq 4$, results in a word. $(\text{fdeletepref})$

$^7$ $|x| \leq 4$ means that the length of the ‘prefix’ or the ‘suffix’ must not exceed 4 characters.

$^8$ These templates are costly to compute since for each word $W$ we get a template of the simpler type (the templates except number 8 and 16). To speed up the learning process a threshold $n$ is given as an argument to the learner which restricts the use of these templates to the $n$ most frequent words in $\text{bigwordlist}$ (i.e. the first $n$ words in $\text{bigwordlist}$). This is the reason $\text{bigwordlist}$ is sorted.
15. Change the most likely tag from X to Y if adding the prefix x, |x| ≤ 4, results in a word. (*faddpref*)

16. Change the most likely tag from X to Y if word W ever appears immediately to the left/right of the word. (*fgoodleft*) vs. (*fgoodright*)

Note that the only difference between the first eight and the second eight templates is that in the latter the template changes a tag X to another tag Y while in the first eight templates the new tag will be Y regardless of the current tag.

The lexical rules produced by the lexical learning module are then used to initially tag the unknown words in the contextual training corpus. This will be described in detail below.

### 3.3 Learning contextual rules

Once the tagger has learned the most likely tag for each word found in the manually annotated training corpus and the method for predicting the most likely tag for unknown words, contextual rules are learned for disambiguation. The learner discovers rules on the basis of the particular environments (or the context) of word tokens.

The contextual learning process needs an initially annotated text. The input to the initial state annotator is an untagged corpus, a running text which is the second half of the manually annotated corpus where the tagging information of the words is removed. The initial state annotator needs a list, a so called *traininglexicon*, which consists of a list of words with a number of tags attached to each word. These tags are the tags that are found in the first half of the manually annotated corpus (the ones used by the lexical module). The first tag is the most likely tag for the word in question and the rest of the tags are in no particular order.

![Table](Word Tag \_1 \_Tag \_2 \ldots \_Tag \_n)

With the help of the *traininglexicon*, the *bigbigramlist* (the same as used in the lexical learning module, see above) and the lexical rules, the initial state annotator assigns to every word being in the untagged corpus the most likely tag. It tags the known words, i.e. the words occurring in *traininglexicon*, with the most frequent tag for the word in question. The tags for the unknown words are computed using the lexical rules: each unknown word is first tagged with NN or NNP and then each of the lexical rules are applied in order. The reason why *bigbigramlist* is necessary as input is that some of the triggers (nr. 8 and 16) in the lexical rules are defined in terms of this list. The annotated text thus obtained is the initial *temporary running text* $RT_0$\(^\ast\) which serves as input to the contextual learner.

Transformation-based error-driven learning is used to learn contextual rules in a similar way as in the lexical learner module. The input to the contextual learner is the second half of the manually annotated corpus (i.e. the goal corpus), the initial temporary corpus $RT_0$ and the *traininglexicon* (the same one as above). First, the learner generates the set of all permissible rules PR from all possible instantiations of all the predefined contextual templates. Note that PR in the contextual module and PR in the lexical module are different sets of rules, since the two modules have different transformation templates. Here, the triggers of the templates for the rules usually depend on the current context. The following are the triggers of the contextual transformation templates:

---

\(^\ast\) These triggers do not check if the word before/after the current word is a particular word (in the current context), instead they check if the current word ever can be preceded/followed by a particular word in some context.

\(^\ast\) $RT_0$ was called $TC_0$ in the general description of TEL above.
1. The preceding/following word is tagged with Z. (PREVTAG/NEXTTAG)
2. One of the two preceding/following words is tagged with Z. (PREV1OR2TAG/NEXT1OR2TAG)
3. One of the three preceding/following words is tagged with Z. (PREV1OR2OR3TAG/NEXT1OR2OR3TAG)
4. The preceding word is tagged with Z and the following word is tagged with V. (SURROUNDTAG)
5. The preceding/following two words are tagged with Z and V. (PREVBIGRAM/NEXTBIGRAM)
6. The word two words before/after is tagged with Z. (PREV2TAG/NEXT2TAG)
7. The current word is Z. (CURWD)
8. The preceding/following word is W. (PREVWD/NEXTWD)
9. One of the preceding/following words is W. (PREV1OR2WD/NEXT1OR2WD)
10. The word two words before/after is W. (PREV2WD/NEXT2WD)
11. The current word is Z and the preceding word is V. (LBIGRAM)
12. The current word is V and the following word is Z. (RBIGRAM)
13. The current word is V and the preceding/following word is tagged with Z. (WDPREVTAG/WDNEXTTAG)
14. The current word is V and the word two words before/after is tagged with Z. (WDAND2BFR/WDAND2TAGAFT)

For all rules in PR for which the trigger condition is met the scores on the temporary corpus RT<sub>0</sub> are computed. It picks the rule with the highest score, R<sub>1</sub>, which is then put on the output list. Then the learner applies R<sub>1</sub> to RT<sub>0</sub> and produces RT<sub>1</sub>, on which the learning continues. The process is reiterated putting one rule (the one with the highest score in each iteration) on the output list<sup>11</sup> in each step until learning is completed, i.e. no rule achieves a score higher than some predetermined threshold value. A higher threshold speeds up the learning process but reduces the accuracy because it may eliminate effective low frequency rules.

If R is a rule in PR, the score for R on RT<sub>i</sub> is computed as follows. For each word in RT<sub>i</sub> the learner computes the score for R on this word. Then the scores for all words in RT<sub>i</sub> where the rule is applicable are added and the result is the total score for R. The score is easy to compute since the system can compare the tags of words in RT<sub>i</sub> with the correct tags in the goal corpus (the texts are the same). If R is applied to the word w, thereby correcting an error, then the score for w is +1. If instead an error is introduced the score for w is -1. In all other cases the score for w is 0. Thus, the total score for R is

\[
\text{score}(R) := \text{number of errors corrected} - \text{number of errors introduced}.
\]

There is one difference compared to the lexical learning module, namely the application of the rules is restricted in the following way: if the current word occurs in the traininglexicon but the new tag given by the rule is not one of the tags associated to the word in the traininglexicon, then the rule does not change the tag of this word.

The rules produced by the contextual learning module together with the lexical rules and several other files can then be used as input to the tagger to tag unannotated text, as will be described in the next section.

<sup>11</sup> Recall that the output list is ordered.
3.4 Tagging new texts

To execute the tagger several files are used: lexicon, an unannotated corpus which will be tagged by the tagger, bigbigramlist, lexical rule file and contextual rule file. For calling the tagger, the following command has to be written:

tagger LEXICON UNTAGGEDCORPUS BIGBIGRAMLIST LEXICALRULEFILE CONTEXTUALRULEFILE > Outputfile

bigbigramlist is the same as above, the lexical and contextual rule files are the ones computed by the lexical and the contextual learner modules. The lexicon is built in the same way as traininglexicon above, the only difference being that the whole manually tagged corpus is used. The bigbigramlist is needed to check the trigger conditions for the lexical rules.

First, the tagger tags all known words (words occurring in the lexicon) with the most frequently occurring tag for the word in question. Then, the tags for the unknown words are computed using the lexical rules: each unknown word is first tagged with NN or NNP and then each of the lexical rules are applied in order. This completes the lexical part of the tagging. Then, the contextual rules are applied to the whole corpus in order with the same restriction as in section 3.3. The result is the annotated text.

3.5 Some practical information

The software is distributed freely and is available from

‘ftp.cs.jhu.edu/pub/brill/Programs/RULE_BASED_TAGGER’

Information about how to compile the programs and how to train and test the tagger is found in README files in the directory Docs.

Note that all text files, both annotated and unannotated used for training and testing must be tokenised which includes removing the punctuation from words and placing one sentence per line. Tagged text is of the form word/tag. For example, annotated text taken from the Hungarian test corpus has the following form,

De/KOT kemelnyselge/FN_PSe3 volt/IGE_Me3 a/DET legnagyobb/MN_FOK ./.

while unannotated text taken from the same corpus is as follows,

De/kemelnyselge volt a legnagyobb .

The normalisation of the Hungarian training and test corpus will be described in section 5 but first a presentation of the Hungarian language follows.
4 The Hungarian language

Hungarian, also called Magyar, traditionally belongs to the Ob-Ugric languages (e.g., Khanty and Mansi) of the Finno-Ugric branch of Uralic. Hungarian is the official language of the Republic of Hungary, and has approximately fifteen million speakers, of which four million reside outside of Hungary.

In this section a description of Hungarian phonology, morphology and syntax follows. The subsections are based on Benkő & Imre (1972), Rácz (1968), Olsson (1992) and Abondolo (1992).

4.1 Phonology

Hungarian has a rich system of vowels and consonants.

The vowel inventory consists of 14 phonemes of which one can distinguish 5 pairs, consisting of short and long counterparts; these are i - í, o - ó, ö - Ő, u - ú, ü - ı. The remaining four are e - é and a - á. Short vowels, if they are marked, take an umlaut (¨), while long vowels are indicated by an acute (´) or with a double acute accent (´´) which is a diacritic unique to Hungarian. Long vowels are usually somewhat tenser than their short counterparts with two exceptions; e is low while é is higher mid and a is low whereas á is lower mid and slightly rounded (Abondolo, 1992). Vowel length is independent of prosodic factors such as stress.

The vowels may be interconnected through the laws of vowel harmony which means that suffixes, which may assume two or three different forms, usually agree in backness with the last vowel of the stem. In other words, front vs. back alternatives of suffixes are selected according to which vowel(s) the stem contain(s) (Benkő & Imre, 1972). The vocalism of stems, classified by Abondolo (1992), is inherently back for all stems containing at least one back vowel and for most verbs with the sole vowel i or í. For all other stems the vocalism is front. In regard to vowel harmony, i and í are neutral and can be used with either front (high) or back (low) vowels. Harmony causes the following alternations among suffix combinations: a/e (-ban/-ben - 'in'), á/é (-nál/-nél - ‘at’), ó/Œ (-ból/-bŒl - ‘from’), u/ü (-ül/-ül - ‘for, by’) and o/e/ö (-hoz/-hez/-höz - ‘to’).

The vowels may show even paradigmatic alternations as long and short vowels (é vs. e and á vs. a, see the example below) alternate in some stems (Benkő & Imre, 1972).

Example B

<table>
<thead>
<tr>
<th>tehén</th>
<th>tehen-et</th>
<th>fa</th>
<th>fá-t</th>
</tr>
</thead>
<tbody>
<tr>
<td>cow:NOM</td>
<td>cow-ACC</td>
<td>tree:NOM</td>
<td>tree-ACC</td>
</tr>
</tbody>
</table>

‘cow’ ‘tree’

When building a Hungarian corpus it is usual to delete the accent and the umlaut from the vowel and mark vowel length as well as the umlaut by numerals which follow the vowel: 1 denotes the acute accent (e.g. ó -> o1), 2 the umlaut (ü -> u2) and 3 the double acute accent (e.g. Ő -> Ő3). This notation can be useful when automatically deriving rules from a corpus because of the paradigmatic alternations of the long vs. short vowels (see example b).

There are totally 25 consonants which can be determined according to the manner and the place of articulation, voicing and quantity. The consonants are shown in the table below.
Table A. The Hungarian consonant chart, given with regular orthographic symbols. Phonetic values are given in square brackets.

<table>
<thead>
<tr>
<th></th>
<th>labial/labiodental</th>
<th>dental/alveolar</th>
<th>palatal</th>
<th>velar</th>
<th>glottal</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>-voice</td>
<td>+voice</td>
<td>-voice</td>
<td>+voice</td>
<td>-voice</td>
</tr>
<tr>
<td>Stops</td>
<td>p</td>
<td>b</td>
<td>t</td>
<td>d</td>
<td>ty [t']</td>
</tr>
<tr>
<td>Affricates</td>
<td>c [ts]</td>
<td>dz</td>
<td>cs [c']</td>
<td>dzs</td>
<td>j˘</td>
</tr>
<tr>
<td>Fricatives</td>
<td>f</td>
<td>v</td>
<td>sz [s]</td>
<td>z</td>
<td>s˘ [s']</td>
</tr>
<tr>
<td>Nasals</td>
<td>m</td>
<td>n</td>
<td>ny</td>
<td></td>
<td>[n']</td>
</tr>
<tr>
<td>Laterals</td>
<td>l</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tremulants</td>
<td>r</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Glides</td>
<td>j, ly</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Consonant length is distinctive and is independent of vowel length and of prosodic factors such as stress. Each consonant can be pronounced short or long, where the last mentioned has almost double the length of short consonants and is written by doubling the letter (gg), or the first element of a digraph (ggy) (Benkő & Imre, 1972). Many of the long consonants occur at morpheme boundaries or root-finally in foreign vocabulary (Abondolo, 1992).

Assimilation is either full or partial and can be indicated by orthography. In section 4.2 on part of speech features, the most important morpheme specific assimilation rules will be presented.

4.2 Morphology

Hungarian is basically agglutinative, i.e. grammatical relations are expressed by means of affixes. For understanding the function of different affixes and how they interact the following section will give an overview of these for different parts of speech. The types of homography will also be described under the main categories.

4.2.1 Articles

The articles [DET] are invariable for number, person, gender and case. The indefinite article is egy, while the definite article has two forms a and az, where the first is used before consonants and the latter before vowels, similar to English indefinite articles.

With regard to homography, the form of the definite article az can also be a demonstrative pronoun and the indefinite article can be homonymous to the numeral ‘one’ (Pajzs, 1996).

4.2.2 Nouns

Every Hungarian noun [FN] may be analysed as a stem followed by three positions in which inflectional suffixes can occur. Thus, nouns are inflected for number, person (possessor) and case, with the relevant suffixes attached in that order (Abondolo, 1992). Any or all of the three inflectional suffixes may be occupied by a zero suffix which

12 j and ly are pronounced alike.
13 Within brackets [] are those PoS tags that are used in the Hungarian corpus.
denotes either singular number (first position), absence of possessor (second position) or nominative case (third position) (Abondolo, 1987). Thus we have:

**Example C**

<table>
<thead>
<tr>
<th>gyereke-Ø-m-en</th>
<th>gyereke-k-Ø-en</th>
<th>gyereke-i-m-Ø</th>
</tr>
</thead>
<tbody>
<tr>
<td>child-Ø-1POSS-SUPESS</td>
<td>child-PL-Ø-SUPESS</td>
<td>child-PL-1POSS-Ø</td>
</tr>
<tr>
<td>‘on my child’</td>
<td>‘on children’</td>
<td>‘my children’</td>
</tr>
</tbody>
</table>

There is no grammatical gender. The personal pronoun \(OE\) means both ‘he’ and ‘she’.

Describing Hungarian, many authors, not without reason, ‘forget’ to mention something about the complicated system of noun stem alternation. Here, an outline of this system based on the summary by Abondolo (1987) is presented.

Noun stems may end in either a consonant or a vowel. All stems with final \(a\) or \(e\) are lengthened (\(á\) vs. \(é\)) before most suffixes, whether derivational or declensional.

**Example D**

<table>
<thead>
<tr>
<th>lámpa =&gt; lámpá-m</th>
<th>kefe =&gt; kefé-m</th>
</tr>
</thead>
<tbody>
<tr>
<td>lamp:NOM</td>
<td>brush:NOM</td>
</tr>
<tr>
<td>‘lamp’</td>
<td>‘brush’</td>
</tr>
</tbody>
</table>

There also exists a very special stem form, called ‘oblique stem’, which occurs with several nouns. This oblique stem form differs from the nominative singular in a way that there is present a stem-final \(a\) or \(e\), or there is absent the stem-penultimate \(o\), \(ö\) or \(e\), or both. Example e illustrate this where fal ‘wall’, dal ‘song’, gyomor ‘stomach’, nyomor ‘misery’, sátor ‘tent’ and mámor ‘rapture’ (Abondolo, 1987).

**Example E** Different nouns have different inflectional patterns based on their oblique stems

<table>
<thead>
<tr>
<th>Stem</th>
<th>present (a) normal</th>
<th>absent (o) normal</th>
<th>absent (o) and present (a) normal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nominative</td>
<td>fal</td>
<td>dal</td>
<td>gyomor</td>
</tr>
<tr>
<td>Oblique</td>
<td>fal-a</td>
<td>dal-</td>
<td>gyomor</td>
</tr>
<tr>
<td>Accusative</td>
<td>fala-t</td>
<td>dal-t</td>
<td>gyomr-ot</td>
</tr>
</tbody>
</table>

There are words whose oblique stems not only have final \(a\) or \(e\) but also \(v\) instead of \(u\), e.g. falu ‘village’ whose oblique stem is falva-, as in the form falva-k ‘villages’.

Nominative stems with the long vowels \(á\) and \(é\) also change to short \(a\) vs. \(e\) in the penultimate position when becoming oblique stem, e.g. madár ‘bird’, oblique stem madara-, thus madara-k ‘birds’. This phenomenon may cause problem when automatically identifying stems.

### 4.2.2.1 Number

The category number is realised as singular and plural. There are two plural suffixes. The suffix -\(k\) is preceded by an epenthetic vowel after a consonant final stem (Olsson, 1992).

**Example F**

<table>
<thead>
<tr>
<th>SINGULAR PLURAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>‘university’</td>
</tr>
<tr>
<td>‘student’</td>
</tr>
</tbody>
</table>

The other plural suffix is -\(i\), which is used only when person suffixes are present, as in example c and example g.
4.2.2.2 Person/Possession

Possession is usually indicated with a personal suffix on the possessed noun. The forms vary for number and person, as shown in Appendix A under possessives. If there is a chain of possessors, the last possessor, closest to the head, takes a dative case marker -nak/-nek in addition to the possessive suffix. Consider the following example.

Example H

<table>
<thead>
<tr>
<th>szóformája</th>
<th>nyelvi jelentés</th>
</tr>
</thead>
<tbody>
<tr>
<td>az apá-m</td>
<td>the father-1SG.POSS</td>
</tr>
<tr>
<td>barát-já-nak</td>
<td>friend-3SG.POSS-DAT/GEN</td>
</tr>
<tr>
<td>a könyv-e</td>
<td>the book-3.SG.POSS</td>
</tr>
</tbody>
</table>

The suffix -já- marks possession by apám ‘my father’ and -nak signals the pending possessive ending -e (Campbell, 1991).

Nouns consisting of a stem with a possessive ending followed by a cases suffix are in most cases homonymous. For example, the word fej+é+nek with the suffix -é- as a possessive suffix means ‘to the possession of his head’, while the same word with -é- as a paradigmatic alternation to the vowel e means ‘to his head’ (Pajzs, 1996).

4.2.2.3 Case

Hungarian has a complex case system involving 16 to 24 distinct forms to mark that an NP bears some identifiable grammatical or semantic relation to the rest of the sentence. The case suffixes may be classified into two groups, non-local and local. The non-local cases express primary syntactic or adverbial functions, such as subject, direct and indirect object, possessor or instrument. The local cases show concrete spatial and kinetic conditions such as interior vs. exterior, stationary vs. moving (Abondolo, 1987).

There are different assumptions about the exact number of case suffixes. I count 19 and the names of the cases and the forms (allomorph) of the suffixes are given in table b with examples (ház ‘house’, öt ‘five’). Unfortunately, there is no space for explaining the function of each case but hopefully the examples illustrate their functions.

Case suffixes are the same both in singular and in plural. The plural suffix always precedes the case suffix, as the examples below show:

Example I

<table>
<thead>
<tr>
<th>szóformája</th>
<th>nyelvi jelentés</th>
</tr>
</thead>
<tbody>
<tr>
<td>a háza-k-ban</td>
<td>the house-PL-INESS</td>
</tr>
<tr>
<td>a háza-i-m-ban</td>
<td>the house-PL-1SG-INESS</td>
</tr>
<tr>
<td>‘in the houses’</td>
<td>‘in my houses’</td>
</tr>
</tbody>
</table>

Assimilation at juncture takes place in instrumental and translative case which may cause problems in automatic tagging systems. The initial -v- in the instrumental suffix -val/-vel and the translative suffix -vá/vé is assimilated to a preceding consonant (Olsson, 1992).
The case suffixes may also occur as stems and take personal suffixes if not postposed to a noun. They are then usually regarded as pronouns or adverbs in traditional Hungarian grammar (see sections 4.2.3 and 4.2.8).

### Table B. The Hungarian case system, listed with each allomorph of the case suffix in singular and plural and exemplified by the words ház 'day', and öt 'five'

<table>
<thead>
<tr>
<th>Case [Tag]</th>
<th>Suffixes</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nominative [NOM]</td>
<td>-</td>
<td>ház house</td>
</tr>
<tr>
<td>Accusative [ACC]</td>
<td>-t, -ot, -et, -öt</td>
<td>házat house</td>
</tr>
<tr>
<td>Dative-genitive [DAT]</td>
<td>-nak, -nek</td>
<td>háznak of the house</td>
</tr>
<tr>
<td>Instrumental [INS]</td>
<td>-(V)al, -(V)el</td>
<td>házzal with the house</td>
</tr>
<tr>
<td>Essive-modal [SOC]</td>
<td>-stul, -stül</td>
<td>házastul with the house and its parts</td>
</tr>
<tr>
<td>Translative [FAC]</td>
<td>-(V)á, (V)é</td>
<td>házzá into a house</td>
</tr>
<tr>
<td>Causal-final [CAU]</td>
<td>-ért</td>
<td>házért for the house</td>
</tr>
<tr>
<td>Illative [ILL]</td>
<td>-ba, -be</td>
<td>házba into the house</td>
</tr>
<tr>
<td>Sublative [SUB]</td>
<td>-ra, -re</td>
<td>házra onto the house</td>
</tr>
<tr>
<td>Allative [ALL]</td>
<td>-hoz, -hez, höz</td>
<td>házhoz to the house</td>
</tr>
<tr>
<td>Inessive [INE]</td>
<td>-ban, -ben</td>
<td>házban in the house</td>
</tr>
<tr>
<td>Supressive [SUP]</td>
<td>-n, -on, -en, -ön</td>
<td>házon on the house</td>
</tr>
<tr>
<td>Adessive [ADE]</td>
<td>-nál, nél</td>
<td>hánál at the house</td>
</tr>
<tr>
<td>Elative [ELA]</td>
<td>-ból, -ből</td>
<td>házból out of the house</td>
</tr>
<tr>
<td>Delative [DEL]</td>
<td>-ről, röl</td>
<td>házról from (top of) the house</td>
</tr>
<tr>
<td>Ablative [ABL]</td>
<td>-től, től</td>
<td>háztól from (nearby) the house</td>
</tr>
<tr>
<td>Terminative [TER]</td>
<td>-ig</td>
<td>házig as far as the house</td>
</tr>
<tr>
<td>Formal [FOR]</td>
<td>-ként</td>
<td>házként as a house</td>
</tr>
<tr>
<td>Temporal [TEM]</td>
<td>-kor</td>
<td>ötkor at five</td>
</tr>
</tbody>
</table>

### 4.2.3 Pronouns

The use of personal pronouns [NM] is not frequent in Hungarian because it is a pro-drop language. They basically have two cases: nominative and accusative. The singular forms in nominative case can be derived from the corresponding singular by adding the plural suffix -k (the same as the plural suffix for nouns) to the singular stem, e.g. (I) +k. For the other plural forms there is no such simple connection (we and ‘you’).

The personal pronouns in first and second person singular in accusative occur usually without the accusative suffix -t. (engem ‘me’, téged ‘you’). The first and second plural forms in accusative are, on the other hand, constructed as nominative form + corresponding possessive suffix + accusative marker -t, i.e. the same marker for possessive suffixes and for the accusative for nouns, e.g. mi+nk+et ‘us’ and ti+tek+et ‘you’. The third person in accusative can be derived as nominative + (plural ending) + accusative ending, thus Ő+t her/him Ő+k+et ‘them’.

By adding the enclitic (personal) markers to the case endings of nouns, listed in table b, oblique forms are made. These correspond to prepositional phrases in English, and are often regarded as pronouns in different cases, or as adverbs because of their adverbial function in the sentence. The enclitic markers with examples are shown in table c.

---

14 The accusative case ending in certain constructions may be zero (-Ø-) if the object is a noun with a possessive personal endig, e.g. eladom a házam/házamat ‘I sell my house.’.
15 The reason for marking the genitive and the dative cases as the same is, that dative may mark not only the indirect object but also the possessor.
Table C. The enclitic markers for Hungarian pronouns

<table>
<thead>
<tr>
<th>Singular</th>
<th>Plural</th>
<th>Examples - SG</th>
<th>Examples - PL</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 -m</td>
<td>unk, -unk</td>
<td>nek-em 'to me'</td>
<td>nek-unk 'to us'</td>
</tr>
<tr>
<td>2 -d</td>
<td>tok, -tek</td>
<td>nek-ed 'to you'</td>
<td>nek-tek 'to you'</td>
</tr>
<tr>
<td>3 -i, -e</td>
<td>-ik, -ük</td>
<td>nek-i 'to him/her'</td>
<td>nek-ik 'to them'</td>
</tr>
</tbody>
</table>

**Reflexive pronouns** [NM] consist of the word mag 'pit, nucleus’ and a possessive personal suffix, listed in Appendix A. Besides the reflexive function they also have a non-reflexive function to emphasise the personal pronouns. In reflexive function the accusative ending -at is ‘obligatory’, except the first and second person where the nominative form is common.

**Possessive pronouns** [NM] which serve to express possession have the following forms:
- enyém ‘mine’
- miénk ‘ours’
- tied ‘yours’
- tietek ‘yours’
- övé ‘his/hers’
- övék ‘theirs’

Possessive pronouns cannot stand together with the noun head (the possessed entity). When the noun head is present a personal pronoun is used. Thus, the use of double markers on the pronoun and the noun head at the same time is not allowed in Hungarian.

**Example K**

enyém volt a könyv a(z) (én) könyv-em
‘It was my book.’ ‘my book’

The system of **demonstrative pronouns** [NM] consists of two categories; pronouns with front vowel mean ‘near’ in contrast to pronouns with back vowel which mean ‘far’, e.g. ez/az ‘this/that’. Case endings are usually added to the pronouns and show both regressive (ez-nek => en-nek) and progressive (az-val => az-zal) assimilation. The demonstrative pronoun cannot take enclitic markers or possessive suffixes and takes the same position as the head nominal.

**Example L**

ezek mögött a problémá-i-d mögött
‘behind these problems of yours’

The **interrogative pronouns** [KSZ] are based on the stems ki ‘who’, mi ‘what’ and hol ‘where’. Their derivatives are different case suffixes as in the case of the demonstrative pronouns.

Other pronouns, e.g. relative and indefinite pronouns [NM] are compounds of the interrogative pronouns. **Relative pronouns** are derivable from interrogative pronouns which is done by the prefix a-. It has personal and non-personal counterparts and singular and plural forms as well as full declension. They follow the pattern of Hungarian nouns in case and number.

One can describe other type of pronouns in a similar way, as below.

<table>
<thead>
<tr>
<th>Pronouns</th>
<th>Prefix</th>
<th>Pronominal stems</th>
</tr>
</thead>
<tbody>
<tr>
<td>Relative</td>
<td>a-</td>
<td>+ ki ‘who’</td>
</tr>
<tr>
<td>Indefinite</td>
<td>vala-</td>
<td>+ mi ‘what’</td>
</tr>
<tr>
<td>Negative</td>
<td>se(n/m)</td>
<td>+ milyen ‘what kind’</td>
</tr>
<tr>
<td>Selective</td>
<td>bár- / akár-</td>
<td>+ mely ‘which’</td>
</tr>
<tr>
<td>General</td>
<td>minden</td>
<td>+ mennyi ‘how many’</td>
</tr>
</tbody>
</table>

21
Each pronominal prefix can be added to the pronominal stems. Thus, the indefinite pronoun *valami* means ‘something’, *valaki* ‘somebody’, *valahol* ‘somewhere’, the selective pronoun *bárki* means ‘anyone’, *bármi* ‘anything’ and so on. The above mentioned pronouns also take case- and plural suffixes, e.g. *valamiben* ‘in something’.

### 4.2.4 Adjectives

Adjectives [MN] can be used as nouns and are then declined fully, i.e. plural and case endings can be added to adjectives. There is also a special case only used by adjectives: the modal-essive with the forms -*en*-/*an*-/*on* and -*leg*, e.g. *gazdag-on* ‘in a rich way’.

Comparative is formed by adding a (harmonic vowel) + -*bb* to the stem. Some comparative forms are suppletive, such as *sok* ‘many’ vs. *több* ‘more’. Superlatives are formed by adding the prefix *leg* to the comparative form, e.g. *rossz* ‘bad’, *rossz+abb* ‘worse’, *leg+rossz+abb* ‘worst’.

In Hungarian, adjectives in attributive position precede their head nouns and do not agree with them. On the other hand, adjectives in predicative position agree in number with the subject.

*Example M*

<table>
<thead>
<tr>
<th>a szép virág-ok</th>
<th>a virágok szép-ek</th>
</tr>
</thead>
<tbody>
<tr>
<td>the beautiful flower-PL</td>
<td>the flower-PL beautiful-PL</td>
</tr>
<tr>
<td>‘the beautiful flowers’</td>
<td>‘The flowers are beautiful.’</td>
</tr>
</tbody>
</table>

### 4.2.5 Numerals

Numerals [SZN] precede nouns and do not agree with them. When standing without the head noun they carry the same case endings as the nouns, pronouns or adjectives, e.g. *őt-ért* ‘for five’.

### 4.2.6 Verbs

Hungarian verbs [IGE] may be analysed as a stem, followed by a tense/mood suffix, followed by a person-and-number suffix, see example n. The morphological manifestations of the tense/mood and person/number system are interconnected and will be discussed together. In Appendix A under the Verb [IGE] category all allomorphs of different tense/mood and person/number categories are listed. Below just a few examples of different paradigms follow.

*Example N*

<table>
<thead>
<tr>
<th>ír-Ø-om</th>
<th>ír-t-am</th>
</tr>
</thead>
<tbody>
<tr>
<td>write-PRS-1SG:DEF</td>
<td>write-PST-1SG:DEF</td>
</tr>
<tr>
<td>‘I write’</td>
<td>‘I wrote’</td>
</tr>
</tbody>
</table>

In verb conjugation the personal suffixes play a central role because free pronominal subjects are only present when there is emphasis on the person, as in example n above. Personal suffixes express first, second and third person not only with pronouns, nouns and postpositions to mark personal relations but also with verbal personal endings and on non-finite forms of the verb.

The verbal paradigm is split into two conjugations according to the definiteness of the complement of the verb - the direct object. Thus, each person suffix refers not only to the person and number of the subject, but also to the person or the definiteness (though not the number) of the object. The object is definite if it is a proper name, a noun with a
definite article, a noun with a personal ending or a personal pronoun in the third person. Other pronouns in object position take verbs in indefinite conjugation (Benkő & Imre, 1972). Thus, there are two first person singular suffixes in the non-past form of the verb ír ‘write’: -k is used with an indefinite direct object and -m is used with definite objects. Note that both suffixes also refer to the first person singular noun.

**Example O**

<table>
<thead>
<tr>
<th>ír-ok</th>
<th>egy könyv-et</th>
<th>ír-om a könyv-et</th>
</tr>
</thead>
<tbody>
<tr>
<td>write-1SGINDEF a book-ACC</td>
<td>write:1SGDEF the book-ACC</td>
<td></td>
</tr>
</tbody>
</table>

Hungarian basically distinguishes between two tense features; past and non-past. The present tense suffix is zero (-Ø-) while past tense is marked by the -t- suffix. Note, that -t also marks the accusative case for noun, pronouns, etc.

**Example P**

<table>
<thead>
<tr>
<th>ír-Ø-ja</th>
<th>a könyvet</th>
</tr>
</thead>
<tbody>
<tr>
<td>write-PRS-3SGDEF the book-ACC</td>
<td></td>
</tr>
<tr>
<td>‘He/she is writing the book.’</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ír-t-a</th>
<th>a könyv-et</th>
</tr>
</thead>
<tbody>
<tr>
<td>write-PST-3SGDEF the book-ACC</td>
<td></td>
</tr>
<tr>
<td>‘He/she was writing the book.’</td>
<td></td>
</tr>
</tbody>
</table>

The future tense is made with the auxiliary ‘fog’ with different personal endings + infinitive (-ni), as in fogok/fogom adni ‘I’ll give’ depending on whether the complement is indefinite or definite (Campbell, 1991). The present tense used with the verb particle meg- can also be used in future tense, e.g. megírom a könyvet ‘I’ll write the book’.

There are three mood categories: indicative, subjunctive, which also functions as imperative, and conditional. In *indicative* mood, as it was mentioned above, the present and past tenses are made by means of personal endings for every person and number, and for definite and indefinite complement.

In imperative/subjunctive mood the marker is -j. In verbs in which the root ends in t, the j suffix of the resulting ‘tj’ is phonetically realised as ‘ss’ and this is sometimes indicated in the orthography: más+j => mássz ‘Climb!’ and mos+ja => mossa ‘She washes it’ (Benkő & Imre, 1972).

The conditional form is made by the marker -n- followed by a harmonic vowel, e.g. adnám ‘I’d give’ (definite object) and kérnék ‘I would ask’ (indefinite object) (Campbell, 1991).

The rich marking system is complicated because of allomorphic variation where some of the allomorphs are not phonologically indicated, but rather to avoid homonymy. For example, the first person singular past tense form lát-ta-m ‘I saw’ is used both with indefinite 3rd person object, no object and with the definite object. The reason for this, according to Olsson (1992) is that the indefinite form would otherwise be identical with the third person plural in past tense with indefinite object lát-ta-k ‘they saw’ because of the personal suffix -k with indefinite objects.

Selection of the personal suffix is governed by phonological, lexical and tense/mood factors. There are over twenty distinct suffixes of person because of the two conjugations and in order to avoid homonymy. There are also words which are homonymous between the past tense and the present tense of different verbs, such as vált ‘he became’ and ‘he changes’ (Pajzs, 1996).

Furthermore, verbs in third person singular are often homonymous with nouns in nominative case, e.g. vár ‘he waits vs. ‘castle’. Sometimes even a conjugated form of a verb and an inflected form of a noun are homonymous, as in várnak ‘they wait’ vs. ‘to the castle’ (Pajzs, 1996).
4.2.6.1 Copula

In Hungarian, the copula [IGE] expresses that something exists. It can signal the existence of something, someone, place, and also signifies time, weather, a material, an origin, a cause, or a purpose. The use of the copula expresses even possession, e.g. ‘to have something’ where many other languages have a transitive verb for this type of construction.

The copula like other verbs is conjugated according to person/number, tense and mood. In the case of present indicative third person singular/plural the copula is realised as zero (Ø) if the predicate is nominal, e.g. pronoun, noun or adjective and the predicate expresses a profession, thing, state, quality or characteristic, etc. This phenomenon may cause problems in automatic tagging systems because there is no verb in the sentence and the elements may be considered as an example of a single NP.

If the predicate is an adverbial and signifies place, time, purpose, etc. or expresses possession the form in third person singular is van and in third person plural is vannak.

The negation of the singular form van is nincs ‘is not’ and of the plural form vannak is nincsenek ‘are not’. All other forms of the copula are negated by a preposed nem ‘no/not’, e.g. nem vagyok ‘I am not’.

4.2.6.2 Infinitive

Hungarian infinitives [INF], unlike most European languages, may be inflected for person. The reason for this is functional because often there is no other element in the sentence to mark the person. The suffixes are almost identical to the nominal paradigm, except in the third person both in singular and plural, where there is an i instead of the epenthetic j (see Appendix A).

4.2.6.3 Participles

The present, past and ‘future’ suffixes of the participles [MN] are -ő/-Œ, -t/-tt, and -andő/-endŒ, respectively. The suffixes are added to the verb stem, and follow the rules of vowel harmony. The present participle form (-ő/Œ) is quite productive and is often used as a noun (Campbell, 1991).

Example Q

a dolgoz-ő ember
the work-PART man
‘the working man’

a dolgozó-k
the worker-PL
‘the workers’

The main function of the past participle is to express an antecedent action and the states which result from it. It syntactically often behaves as adjective. Its form is identical with the third person singular past indefinite verb form (-t).

The participles are not used as predicates. Instead, there is a structure consisting of the copula and the verbal adverb with the suffix -va/-ve. The difference between participles and verbal adverbs is that the latter are more closely connected with the finite verb of the sentence than the participle is in time, state or mood. Participles modify the noun head while verbal adverbs modify the verb. The use of the verbal adverb is more limited than the use of the participle. The following examples show the difference between them.
Example R

a szobá-ban ül-Œ gyerek-ek játsza-nak
the room-INESS sit-PRSPART child-PL play-3PL
‘The children sitting in the room are playing.’

a gyerek-ek a szobá-ban ül-ve játsza-nak
the child-PL the room-INESS sit-VERBADV play-3PL
‘The children are playing sitting in the room.’

4.2.6.4 Verbal particles

Hungarian has a very rich system of verbal particles or verb prefixes [IK] which are separable from the stem (Campbell, 1991). They serve to mark direction (le ‘down’, ki ‘out’, etc.), aspect (meg ‘completed’) and to make verbs transitive. They can be combined with many verbs. In many cases there is a different meaning of the verb depending on what particle is attached to it, e.g. átad ‘hand over, pass’, elad ‘sell’, etc.

Verbal particles may have two positions depending on the emphasis within the sentence. In the case of a neutral sentence or in yes or no questions the particle is a prefix attached to the verb.

Example S

Péter ki-megy a szobá-ból ./?
Peter out-go:3SG the room-ELAT
‘Peter leaves the room./Does Peter leave the room?’

The particle follows the verb in questions, in negatives and when any part of the sentence is emphasised.

Example T

Péter nem megy ki a szobá-ból ./?
Peter not go:3SG out the room-ELAT
‘Peter doesn’t leave the room./Doesn’t Peter leave the room?’

Note that in example s and example t the local relation is marked twice within each structure: once with verbal prefix and once with the case (elative) of the noun.

4.2.7 Postpositions

Postpositions [NU] follow the head they refer to and express principally local relations and show a three-way opposition for motion relative to the speaker or other referents. Postpositions may even represent temporal or abstract meaning, e.g. után ‘after’ or ellen ‘against’. They are reduplicated with demonstratives (DEM) as shown the example below.

Example U

az alatt az asztal alatt
DEM under the table under
‘under that table’

The postpositions, like case markers, may occur as stems and take possessive endings, (see Appendix A) and in this form according to traditional Hungarian grammar they are considered as adverbs. This can be problematic for an automatic tagging system because
the stem of these type of adverbs is a postposition hence will be annotated as a postposition [NU], rather than an adverb [HA].

4.2.8 Adverbs

One type of adverb [HA] is the form of case_marker/postposition + personal ending. In some corpus, words consisting of case_marker + personal ending are considered as pronouns, while in another corpus they may be considered as adverbs, as was mentioned in section 4.2.3.

There are also adverbs which are derived from verbal particles expressing local relations though without possessive endings, such as be ‘in’ vs. bent ‘inside’, ki ‘out’ vs. kint ‘outside’. The local relation is marked twice within a structure: the adverb and the case marker on the noun.

Example V

Bent van a szobá-ban.
inside COP:3SG the-room-INESS
‘He/she/it is inside the room.’

4.2.9 Word formation

This section will give a brief overview on composition and derivation of words in Hungarian which can cause problems when automatically analysing texts. This part is based on Benkő & Imre (1972:145-156) where the interested reader may find more information with examples on this topic.

4.2.9.1 Word composition

Compound nouns are very frequent in Hungarian. Two nouns can simply be combined without any formal means, e.g. kávé+ház ‘coffee’+‘house’. Another type of compound nouns is where the first constituent is a participle, e.g. mosó+nő ‘washing’+‘woman’. The participle can also occur as a second member but the word is already substantivised and considered to be a noun, e.g. adó+szedő ‘tax’+‘collector’. These types of words may be confused with adjectives by an automatic tagging system because of their ending, typical to adjectives.

Compound adjectives consist of a noun + adjective. The compound may express similarities, e.g. jég+hideg ‘ice’+‘cold’ or the adjective member limits a certain range of meaning, e.g. adó+mentes ‘tax’+‘free’.

There are also compounds which consist of i) a noun with adverbial ending or adverb + noun, ii) a noun with adverbial ending + non-finite verb form, iii) a noun with possessive suffix + participle, and iv) a noun with adverbial ending or adverb + verb etc. These may also cause problems in automatic morphological analysis.

4.2.9.2 Derivation of words

Unlike most European languages, Hungarian has a very regular system for derivational suffixes where a single suffix corresponds to a significant meaning and its use is regular. One suffix is tied to each important suffix function. Unfortunately, there is no space for describing the whole system with each derivational suffix, so only a few examples of those suffixes which change the category of a word are given.
Derivational suffixes which change the part of speech of a word are very common and productive. Verbs, nouns, adjectives and even adverbs can be further derived. Deverbal verb suffixes, for example express frequentative-iterative, causative and reflexive meaning. Thus, the suffix -hat/-het means 'can, is capable' and 'may, is possible', e.g. kime-het 'he/she may go out'. Denominal verb suffixes are -l, -z, - Kodik/-kedik, - skodik/-skedik, e.g. szolga 'servent' => szolga-l 'serve', társ 'fellow' => társa-lkedik 'converse'. Deaddress verb suffixes are also very productive, such as the suffix - kodik which coincides with the denominal verb suffixes, e.g. ügyes 'skilful' => ügyes-kedik 'behave skilfully'. Deverbal noun suffixes are also very common, especially the suffix -ás/-és as in temet 'bury' => temetés 'burial, funeral', and the suffix -at/-et as in talál 'find, discover, hit' => talál-at 'hit, win'. The suffix -ól-/OE for the name of a profession which is identical with the participial belongs also to this category. The deaddress noun suffix -ság/-ség is a very productive one, e.g. alázatos 'humble' => alázatos-ság 'humbleness'. Adjectives may also be derived from verbs and nouns. Deverbal adjective suffixes among others are -tlan/-tlen, e.g. árt 'harm' => árt-atlan 'harmless', and -ős/OEs, e.g. nyúl 'stretch' => nyúl-ős 'stretchy'. Some examples for denominal adjective suffixes are -s, -talan/-tlan, -i, -ál/-á, -já/-ju. Note that several derivational suffixes may follow each other within one form, e.g. le-ír-hat-atlan 'unwritable'.

4.3 Syntax

In the area of Hungarian syntax, the word order, main types of sentence structure, and agreement will be briefly described.

4.3.1 Word order

In most of the literature, Hungarian is described as a basically free word order language. But the fairly free word order can be said to be on the sentence level where the order of the major constituents is only free with respect to grammatical functions and their cases. The word order is pragmatically oriented with a special position for focused or emphasised constituents before the finite verb. The basic order of the sentence constituents is topic + focus + finite verb + any other items (Abondolo, 1992). Both topicalised and focused elements receive sentence stress. If a sentence contains both then the focused element is more prominent. example w shows some possible word orders for the sentence Péter vesz egy könyvet a boltban 'Peter buys a book at the store'. The constituents of the sentence below (the subject, the verb, the direct object and the adverbial) can be ordered in 24 ways (4!).

Example W

Péter vesz egy könyvet a boltban.
Péter vesz a boltban egy könyvet.
Péter egy könyvet vesz a boltban.
Péter egy könyvet a boltban vesz.
Péter a boltban egy könyvet vesz.
Péter a boltban vesz egy könyvet.

Péter vesz egy könyvet a boltban
Peter buys a book the store:INESS
‘Peter buys a book at the store’
However, the free word order description is inadequate on the phrase level and as soon as interrogative pronouns and/or negated complements are involved. Below, just a few examples are given.

There is a strict word order within the NP. Determiners and demonstrative pronouns always precede the noun head, e.g. in example w, the determiner *a* precedes the noun head *könyvet* ‘book:ACC’. Qualifiers, like adjectives and participles within an NP occur after the determiners and demonstratives, but precede the noun, as in example below.

*Example X*

```
az-ok az első csodálatos karibiai nyári nap-ok
```

‘those first wonderful Caribbean summer days’

Furthermore, qualifiers may themselves be qualified by preceding adverbial complements.

*Example Y*

```
a zöld ház-nál dohányzó fiatal nyelvész-ek
```

‘the young linguists smoking at the green house’

Since Hungarian shares some typological characteristics of SOV languages (e.g. it is postpositional, the attribute precedes the noun, etc.) it is often described as a SOV language. On the other hand, some researchers maintain that Hungarian is partly SOV, partly SVO. However, these word orders are canonical and only represent the dominant order of simple declarative sentences, containing a nominal subject and a nominal object. In the next section, the main types of sentence structure with canonical word order are given. Note that the order of the constituents can be changed according to the information structure.

### 4.3.2 The main types of sentence structure

The rules for the most basic syntactical forms of sentence structure are taken from Benkő & Imre, (1972:86-87) and given below. Grammatical categories which begin with a capital letter represent stem and affix and can be expanded by a subclass with the same name.

1. \[\text{Noun}]_{subject} + \{\text{Adjective + auxiliary_verb}\}_{predicate} \quad \text{where the auxiliary may be zero.}
2. \[\text{Noun}]_{subject} + \{\text{Noun + auxiliary_verb}\}_{predicate} \quad \text{where the auxiliary may be zero.}
3. \[\text{Noun}]_{subject} + \{\text{Copula}\}_{predicate}
4. \{\text{Noun}:case + \text{Copula}\}_{predicate}
5. \{\text{noun:dative + Copula}\}_{predicate} + \{\text{noun:possessive_suffix}\}_{subject}
6. \{\text{Noun}\}_{subject} + \{\text{Verb}\}_{predicate}
7. \{\text{Noun}\}_{subject} + \{\text{Verb + noun:case}\}_{predicate}
8. \{\text{Noun}\}_{subject} + \{\text{Verb + noun:case + noun:case}\}_{predicate}
9. \{\text{Noun}\}_{subject} + \{\text{Verb + Noun + postposition}\}_{predicate}

\[\text{Noun}\]

1. article + noun_stem:base_suffix:case_suffix
   \quad \text{where article may be zero and base_suffix is either a single plural suffix or a possessive personal ending (including the special plural suffix).}
2. article + noun_stem:base_suffix
   \quad \text{where article may be zero and base_suffix is either a single plural suffix or a possessive personal ending (including the special plural suffix).}
3. article + noun_stem:case_suffix
   where article may be zero.
4. noun_stem
5. article + noun_stem:base_suffix + postposition
   where article may be zero and base_suffix is either a plural suffix or a possessive personal
   ending (including the special plural suffix).
6. article + noun_stem + postposition
   where article may be zero and base_suffix is either a plural suffix or a possessive personal
   ending (including the special plural suffix).

[Verb]
1. verb_stem:suffix
   where suffix is personal ending, suffix of time and/or suffix of mood.
2. verb_stem
3. preverb + verb_stem:suffix
   where suffix is personal ending, suffix of time and/or suffix of mood.
4. preverb + verb_stem
5. verb_stem:suffix + auxiliary_verb
   where suffix is personal ending, suffix of time and/or suffix of mood.
6. verb_stem:suffix + auxiliary_verb
   where suffix is personal ending, suffix of time and/or suffix of mood.

Additionally, Hungarian is a pro-drop language, which means that the subject position
of the verb can be left empty. The subject is implicit as the personal endings of the verb
express the first and the second person, and the third person if the context makes clear
who or what the subject is.

4.3.3 Agreement

Generally, syntagmatic relations are marked, if possible, on both members of the
construction which results in redundancy. The following rules illustrate the agreement
patterns (Benko & Imre, 1972).

1. The congruence of noun and substantival pronoun as genitive attribute and qualified
   noun.
   
   Example Z
   
   a te könyv-ed
   DEM you book:2SGPOSS
   a fiú(-nak a) könyv-e
   the boy(-DAT DET) book-3SGPOSS
   ‘your book’ ‘the boy’s book’

2. The demonstrative pronominal dependent ez/az ‘this/that’ agrees in case and in number
   with the noun which it qualifies.
   
   Example AA
   
   eb-böl a könyv-böl azo-k-at a könyv-ek-et
   DEM.PRON-ELAT the book-ELAT DEM.PRON-PL-ACC the book-PL-ACC
   ‘from this book’ ‘those books’

3. Adjectival or numerical attributes are however, not in congruence with the modified
   word.
   
   Example BB
4. As we have seen in section 4.2.6, the verb agrees with the subject in number, person and even with the object, so called object-agreement - the relationship between the person of the subject and the person of the object is marked.

*Example CC*

Péter ír-Ø egy könyv-et. Péter ír-ja a könyv-et.

Peter write-3SGINDEF a book-ACC Peter write-3SGDEF the book-ACC

‘Peter writes a book.’ ‘Peter is writing the book.’

5. Agreement in number between subject and nominal predicate with zero copula.

*Example DD*

A diák-ok szorgalmas-ak.

The student-PL diligent-PL

‘The students are diligent.’

6. The personal endings of the infinitive refers to that part of the sentence which is in the dative and which is the logical subject.

*Example EE*

Péter-nek kell tanulni-a (Én) nek-em kell tanuln-om

Peter-DAT must study-INF3SG (I) me-1SG must study-INF1SG

‘Peter must study.’ ‘I must study.’

7. Verbal particles often agree with the case suffix of the nominal complement of the verb.

*Example FF*

Rá-megy-ek a fi-re.

PART-go-1SG the grass-ILL

‘I step onto the grass.’

In the following section the Hungarian corpora, which have been utilised for training and/or testing Brill’s tagger and their tagset, will be presented.
5 The Hungarian corpora

Two different corpora, both of them opportunistic in the sense that only these were available, were used for training and for testing Brill’s tagger. They were collected and annotated by the Research Institute for Linguistics at the Hungarian Academy of Sciences.

The Hungarian corpus used for training the tagger is the novel *1984* written by George Orwell. It consists of 14034 sentences: 99860 tokens including punctuation marks, 80668 words excluding punctuation marks.

The corpus used for testing the tagger consisted of two texts that were extracted from the large Hungarian ‘hand’ corpus: a poem ‘Az én erdőem’ (‘My forest’), written by István Ágh in the 1970’s, and a fairy tale ‘A három húséges kiráyleány’ (‘The three faithful princesses’) written by Béla Balázs in 1948. They are modern literary pieces, both without archaic words. The poem consists of 397 tokens and the fairy tale of 2110 tokens, both including punctuation marks.

Both corpora, i.e. the test and the training corpus had to be converted into the format that the tagger requires. Thus, the texts were tokenised and altered to one sentence per line before using them on the tagger. The normalisation of the corpora was done with PCBeta, a production system consisting of actions controlled by a set of rules which are created by the user (Brodda, forthcoming). The rules of the normalisation of the corpora are shown in Appendix D.

The *training corpus* (Orwell’s *1984*) as it was taken from the Hungarian Academy of Sciences had the following form before normalisation:

```plaintext
<
/p
>
%Winston
majdnem[HA]
egy[DET]
tealcsze1sze1nyi[MN]+t[ACC]
o2nt[IGE]+o2tt[Me3] #5.3
maga[NM]=maga1+nak[DAT]
,

o2sszeszed[IGE]+te[TMe3]
minden[NM]
lelkiero3[FN]=lelkiere+je1[PSe3]+t[ACC]
,

s[KOT]
lenyel[IGE]+te[TMe3] #5.1
,

@mint[KOT]
egy[DET]
adag[FN]
orvossa1g[FN]+ot[ACC]
.
```

*Figure C. An extract from Orwell 1984 in ‘originally’ annotated version before normalisation*
Each word is written on a new line. In the simplest case, the analysis contains the word token followed by the PoS tag within square brackets ([]). A more complicated analysis appears in the analysis of an inflected word. In this case, the lemma of the token with its PoS tag is given first, followed by one or several inflectional ending(s) with its subtag(s), each preceded by a plus sign (+). If a word begins with a capital letter, the lemma with initial lower-case letter followed by the PoS tag in brackets is given. Then the lemma with the initial capital letter, preceded by the equal sign (=) follows. If the lemma and the stem of the token are not the same, the lemma with its PoS tag is given first, followed by an equal sign. Then the stem of the token is given followed by the inflectional ending(s) with its subtag in square brackets. Each inflectional ending is preceded by a plus sign. Note that part of speech tags and tags for inflectional categories are always enclosed by square brackets ([]).

Long vowels and vowels with umlaut are followed by a numeral: 1 indicates that the vowel is marked with an acute accent, 2 indicates that the vowel is marked with an umlaut and 3 indicates that the vowel is marked with a double acute accent, see also section 4.1.

Punctuation marks are not annotated at all: minor delimiters such as commas are followed by two empty lines and major delimiters, (e.g. full stops) by six empty lines. Headings are marked within arrowhead brackets (<>) where the heading-expression is divided into its components, represented on separate lines preceded by a slash (/) and/or a percent sign.

A line may begin with several marks, e.g. percent sign (‘%’) may indicate that the token is not analysed, (for example foreign proper names are not annotated at all), the ‘at’-sign (@) indicates that there are also several analyses, i.e. the word is ambiguous, etc. Sometimes a number is given after the analysis preceded by a -# or #, which is the identification number of a disambiguation rule, used in the automatic analysis of the corpus.

There were 45 rule files written (see Appendix D) for normalising the original corpus to the form that Brill’s tagger requires. Basically, the rules eliminated items such as (-)#Number, headings, empty lines, alternative analyses and lemmas. Furthermore, the rules annotated punctuation marks, foreign words which were not annotated, and converted the input to the format of Word/Tag1_Tag2_TagN (or just Word/Tag) and one sentence per line. Thus, the output which constitutes the training corpus for PoS tags and for tags denoting inflectional properties (subtags) looks as shown in figure d.

---

Winston/FN majdnem/HA egy/DET tea1sce1sze1nyit/MN_ACC o2nto2tt/IGE_Me3 maga1nak/NM_DAT ./, o2szzeszedte/IGE_TMe3 minden/NM lelkiereje1t/FN_PSe3_ACC ./, s/KOT lenyelte/IGE_TMe3 ./, mint/KOT egy/DET adag/FN orvossa1got/FN _ACC ./.

---

Figure D. An extract from Orwell’s 1984 after normalisation with PoS and subtags

For training the tagger with only part of speech annotated text, tags denoting inflectional properties were removed, as shown in figure e.

---

Winston/FN majdnem/HA egy/DET tea1sce1sze1nyit/MN o2nto2tt/IHE maga1nak/NM ./, o2szzeszedte/IGE minden/NM lelkiereje1t/FN ./, s/KOT lenyelte/IGE ./, mint/KOT egy/DET adag/FN orvossa1got/FN ./.

---

Figure E. An extract from Orwell’s 1984 after normalisation with only PoS tags

In the test corpus, as it was given, each word is written on a new line in conformity with the training corpus. But here, every line consists of four columns where the first one contains a number or a heading mark (#), the second one gives information about whether or not the word is a token or a punctuation mark. The third column consists of the token, followed by a special tag (if there is any), marking if a word begins with a capital letter or
marking the type of punctuation mark. The last column gives the lemma and the PoS and
subtags, divided by back slash (/). Foreign proper names are preceded by #% and are not
annotated. Extracts from the test corpus - from the poem and from the fairy tale - are
shown in figure f and figure g, respectively.

Figure F. An extract from text 2000004009 A1GH ISTVA1N:AZ E1N ERDO3M in the hand corpus in
originally annotated version

Figure G. An extract from text 2000019002 BALA1ZS BE1LA:A HA1ROM HU3SE1GES KIRA1LYLEA1NY in the hand corpus in
originally annotated version
In contrast to the training corpus, the normalisation of the test corpus was considerably easier and faster because of the format and the size of the test corpus. With the help of 14 rules (compared to 45 rules in the case of the training corpus) it was possible to convert the annotated text to the annotated format which Brill’s tagger requires. Mainly, the rules eliminated headings and lemmas, and annotated foreign words, punctuation marks, etc., see Appendix D. For testing the texts on the tagger, the tags (both PoS and subtags) from the tokens were removed. Extracts from the results are shown in figure h and figure i below.

Ko2zel hu1zo1dik e1letemhez , s nem tudom , mekkora , hol ve1gzo3dik , fal nekem o3 , e1lo3 panaszfalom .

In spite of the fact that the tagset of the ‘Orwell’ and the ‘hand’ corpus is the same (see Appendix A), the principles of annotation used in the two corpora are different in some respects. Some categories belong to different levels of annotation. For example, in the ‘hand’ corpus the infinitive verb is annotated with \[IGE][INF] ([Verb][Infinitive] : [IGE] on the top level and [INF] as a subtag, while in the ‘Orwell’ corpus the same item is annotated only as [INF] being on the top level. Another difference lies in that a word consisting of case_ending + personal_ending is tagged as pronoun [NM] in the ‘Orwell’ corpus while the same type of words in the ‘hand’ corpus are tagged as adverbs [HA], sometimes with inflectional tags, sometimes without. Both alternatives are theoretically correct because these types of words behave morphologically as pronouns but syntactically as adverbs. Without guidelines it is difficult to decide which annotation is correct and which is not in a certain context.

In the beginning of this work, the plan was to use the entire correctly annotated ‘hand’ corpus as test corpus (including the poem and the fairy tale) and to automatically compare it with the test corpus annotated by the tagger. Because of the differences and the inconsistencies in the annotation of the training and test corpus this was not possible.
Thus, the largest difficulty with normalisation has been that information about how the annotation was carried out was missing. It would be helpful for future users to be provided with information about the guidelines on which the annotation is based. It would be necessary to suggest principles for the encoding of texts in the same format: the annotation of the ‘hand’ corpus is preferable. The training corpus (Orwell’s *1984*) was tagged by a combination of automatic methods (thus the need for disambiguation rules) and manual tagging though specific information about how this was done is missing from the documentation. Moreover, it is necessary to further develop sets of coding conventions suited for various applications and to maintain compatibility with existing standards as far as possible.
6 Results and evaluation of system efficiency

The tagger has been trained on the same text twice: once with only PoS tags and once with PoS and subtags marking inflectional properties of given words. Both times, the correctly annotated and normalised training corpus was randomly divided into two parts of equal size using a program provided with the tagger. The first half of the manually annotated corpus was used in the lexical learning module, while the second part of the manually annotated corpus was used in the contextual learning module.

Before the lexical learner could be invoked, three files had to be created: bigwordlist, smallwordtaglist and bigbigramlist (see section 3.2). This was done with the software provided with the tagger. The learning module for lexical rules requires these three files and a threshold value. The threshold value was set to 300, meaning that the learner only used bigram contexts, i.e. the neighbour of the current word, among the 300 most frequent words. The default tag for unknown words with an initial capital letter was NNP and for other unknown words NN. Note, that neither NNP nor NP exists in the Hungarian tagset, although these default tags can be changed in the program into the most frequent tag in the Hungarian corpus (FN). The lexical rules are listed in Appendix B.

The next stage was learning contextual rules for improving tagging accuracy, after tagging all words with their most likely tag. That learning process requires (as was mentioned in section 3.3) several files which had to be created: traininglexicon, lexicallytaggedcorpus and manuallytaggedcorpus. The output of the contextual learner, i.e. the contextual rules, are listed in Appendix C.

The total number of lexical and contextual rules with and without subtags is given in table d below. The lexical learner has derived 326 rules based on 31 PoS tags while it has derived 457 rules based on the much larger tagset, consisting of 452 PoS and subtag combinations. Note that if the tagset consists of a large number of frequently occurring tags, the lexical learner necessarily generates more rules simply to be able to produce all these tags. On the other hand, if only PoS tags (excluding subtags) are used the first rules score very high, in comparison with the scores of the first rules based on PoS and subtags, see Appendix B. Another difference is that the score decreases faster in the beginning and slower in the end, compared to the rules based on PoS and subtags, resulting in a large amount of rules, relative to the size of the tagset.

The contextual learner, used to improve the accuracy, derived approximately three times more rules based on 31 PoS tags than it derived from the text annotated with both PoS and subtags. This is somewhat harder to interpret since the output of the contextual learner does not contain scores (although there is an intern scoring function, see section 3.3). It seems reasonable that the contextual rule learner easier finds ‘globally good’ rules, i.e. rules that are better in the long run, since the subtags contain important extra information, for instance about agreement.

The conclusion that can be drawn from these facts and the fact that the test on the training corpus (the entire Orwell’s 1984, see section 6.1) achieved slightly higher precision using subtags, is that it is probably more difficult to derive information from words which are annotated with only PoS tags, than from words whose tags include information about the inflectional categories.

| Table D. The total number of lexical and contextual rules for PoS tags and for PoS with subtags |
|---------------------------------|----------------|----------------|----------------|
| 31 PoS tags                     | 326            | 565            | 891            |
| 452 PoS with subtags           | 457            | 180            | 637            |
The training process\textsuperscript{17} for PoS tags for lexical rules took 18 hours and for contextual rules about twenty-four hours. The lexical training process for PoS and subtags took one week while the contextual learning process took at least twenty-four hours, maybe a lot more (the time was not recorded). The main reason for this difference in time between PoS and PoS and subtags is the fact that the number of all permissible rules in the rule generating process tends to increase if the number of tags is higher.

For evaluating the system efficiency, the tagger has been tested on different types of texts, i.e. on the training corpus and on other texts. All test corpora were annotated with both PoS tags, and PoS together with subtags denoting both the part of speech and the inflectional properties of the word. Thus, the performance of the whole system has been evaluated against a total of three types of texts from different domains, which were presented in section 5. The texts were all literary works, one poem (‘Az én erdőm’, written by István Ágh), one fairy tale (‘A három hiséges királyleány’ by Béla Balázs) and Orwell’s 1984 which constituted the training corpus.

The analysis of the poem and the fairy tale has been manually checked while the analysis of Orwell’s 1984 was checked automatically using the \texttt{diff} function in UNIX which compares two texts for differences.

Precision was calculated for the entire texts, both for PoS tags and PoS with subtags, based on all the tags and the individual part of speech tags. Recall was only calculated for the individual part of speech tags for the poem and the fairy tale. Note that recall is not relevant when considering all the tags, but only when considering specific part of speech categories. Recall of PoS with subtags were not calculated for the poem and the fairy tale since the frequencies of the individual tags were very low. Remember, that precision for a specific PoS tag \(T\) is obtained by

\[
\text{precision of } T = \frac{\text{total number of correctly machine-tagged words with tag } T}{\text{total number of machine-tagged words with tag } T}
\]

and recall is obtained by

\[
\text{recall of } T = \frac{\text{total number of correctly machine-tagged words with tag } T}{\text{total number of words with correct tag } T}
\]

(see also section 2.3). In the following sections the results will be described for each text tested on the tagger. In Appendix E the output of the poem and the fairy tale after testing is shown where the wrongly annotated tokens are underlined and tokens which have a missing inflectional tag are printed in italics.

\section*{6.1 Testing on the training set}

When testing the rules on the training text, Orwell’s 1984, only the contextual rules are used since finallexicon contains all the words of the entire text, see also section 3.4, i.e. all words are considered known.

\begin{itemize}
\item Orwell’s 1984 consists of 99860 tokens.
\item The total number of correctly PoS-tagged tokens is 98432.
\item The total number of tokens which have correct and complete PoS and subtags is 98656.
\item The total number of tokens which have correct PoS tags irrespective of the correctness of the subtags is 98677.
\end{itemize}

\textsuperscript{17} The training was done at the Department of Linguistics, Uppsala University by Klas Prütz using a IBM RISC System/6000 Model 43P.
In this case the difference between the correctly annotated text and the tagger annotated text was automatically checked, thus neither the correct but incomplete subtags nor the precision and recall for specific PoS tags were calculated. In table e below, the precision for both the PoS annotated and the PoS with subtags annotated texts (with and without subtags) are shown.

Table E. Precision for Orwell’s 1984, which constituted the training corpus

<table>
<thead>
<tr>
<th>Orwell</th>
<th>Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td>PoS tags</td>
<td>($98432/99860)\times100 \approx 98.6%</td>
</tr>
<tr>
<td>complete PoS and subtags</td>
<td>($98656/99860)\times100 \approx 98.8%</td>
</tr>
<tr>
<td>correct PoS tags irrespective of the correctness of subtags</td>
<td>($98677/99860)\times100 \approx 98.8%</td>
</tr>
</tbody>
</table>

As mentioned above, these results indicate that the tagger attains slightly better annotation using PoS and subtags, hence better rules, than for only PoS tags. The reason for this could be that marking an inflectional property of a word may give more information about which part of speech the word belongs to.

6.2 Testing on the poem

- The total number of tokens is 397.
- The total number of correctly PoS-tagged words retained by the tagger is 341.
- The total number of words which have correct PoS and subtags but have some subtags that are missing is 356.
- The total number of words which have correct and fully complete PoS and subtags is 352.
- The total number of words with correct PoS tag but not necessarily correct subtag is 366.

Thus, precision can be calculated for the PoS tagged poem and for the PoS and subtagged poem as shown in table f below. The accuracy is lowest (85.9\%) for the PoS tagged version while the annotation with PoS and subtags gives a higher accuracy, namely 88.7\%. When the missing tags for inflectional categories are excluded from consideration, the accuracy increases from 88.7\% to 89.7\%. The best result is given (92.2\%) when only correct PoS tags of the PoS and subtag combinations are considered, i.e. the subtags are not included.

Table F. Precision for poem

<table>
<thead>
<tr>
<th>Poem</th>
<th>with PoS tags</th>
<th>with PoS and subtags</th>
</tr>
</thead>
<tbody>
<tr>
<td>only PoS tags</td>
<td>(341/397)\times100 \approx 85.9%</td>
<td>(366/397)\times100 \approx 92.2%</td>
</tr>
<tr>
<td>excluding missing subtags as correct</td>
<td>-&quot;-</td>
<td>(352/397)\times100 \approx 88.7%</td>
</tr>
<tr>
<td>including missing subtags as correct</td>
<td>-&quot;-</td>
<td>(356/397)\times100 \approx 89.7%</td>
</tr>
</tbody>
</table>

For different parts of speech the precision and recall are given in table g below. In the first column the specific parts of speech are listed. In the second column the first number gives the precision for a specific PoS tag, which is the quotient of the total number of

---

18 After annotation the subtags were removed in order to automatically check the total number of correct PoS tags.
correctly annotated words with that PoS tag and the total number of words annotated with that PoS tag by the tagger. Which part of speech the wrongly annotated words actually belong to is also given. For example, one noun (FN 1), one verb (IGE 1) and one adjective (MN 1) were wrongly annotated by the tagger as infinitive (INF).

In the third column, recall is shown, i.e. the quotient of the total number of correctly annotated words with a specific PoS tag and the total number of words with that intended tag, followed by those categories which the tagger wrongly annotated. For example, one of the 14 infinitives (INF) in the text was analysed by the tagger as a noun (FN 1).

Table G. Precision and recall for different parts of speech in the poem

<table>
<thead>
<tr>
<th>PoS tags</th>
<th>Precision correct_found/retrieved_total</th>
<th>Recall correct_found/intended_total</th>
</tr>
</thead>
<tbody>
<tr>
<td>INF (Infinitive)</td>
<td>0.81 13/16 FN 1 IGE 1 MN 1</td>
<td>0.93 13/14 FN 1</td>
</tr>
<tr>
<td>IGE (Verb)</td>
<td>0.81 42/52 MN 5 FN 4 NU 1</td>
<td>0.72 42/58 FN 10 MN 4 INF 1 SZN 1</td>
</tr>
<tr>
<td>IK (Verbal Particle)</td>
<td>0.80 4/5 NM 1</td>
<td>0.67 4/6 NM 1 HA 1</td>
</tr>
<tr>
<td>FN (Noun)</td>
<td>0.75 70/93 MN 11 IGE 10 NM 1 INF 1</td>
<td>0.86 70/81 MN 6 IGE 4 INF 1</td>
</tr>
<tr>
<td>DET (Determiner)</td>
<td>1.0 25/25 ----</td>
<td>1.0 25/25 ----</td>
</tr>
<tr>
<td>MN (Adjective)</td>
<td>0.76 35/46 FN 6 IGE 4 HA 1</td>
<td>0.67 35/52 FN 10 IGE 5 INF 1</td>
</tr>
<tr>
<td>NM (Pronoun)</td>
<td>0.93 14/15 IK 1</td>
<td>0.88 14/16 FN 1 IK 1</td>
</tr>
<tr>
<td>HA (Adverb)</td>
<td>0.90 28/31 KOT 1 IK 1 KSZ 1</td>
<td>0.88 28/32 KOT 3 MN 1</td>
</tr>
<tr>
<td>SZN (Numeral)</td>
<td>0.50 1/2 IGE 1</td>
<td>1.0 1/1 ----</td>
</tr>
<tr>
<td>NU (Postposition)</td>
<td>1.0 3/3 ----</td>
<td>0.75 3/4 IGE 1</td>
</tr>
<tr>
<td>KOT (Conjunction)</td>
<td>0.88 22/25 HA 3</td>
<td>0.96 22/23 HA 1</td>
</tr>
<tr>
<td>KSZ (Interrogative)</td>
<td>-- 0 ----</td>
<td>-- 0/1 HA 1</td>
</tr>
<tr>
<td>ISZ (Interjection)</td>
<td>-- 0 ----</td>
<td>-- 0 ----</td>
</tr>
<tr>
<td>NNP (Default tag)</td>
<td>-- 0 ----</td>
<td>-- 0 ----</td>
</tr>
</tbody>
</table>

The tagger gave maximum precision and recall for determiners (DET), i.e. neither gave the tagger wrong tags for that category or missed any. All postpositions (NU) which the tagger suggested were correct (maximum precision), but it missed one of the four postpositions. Pronouns (NM), adverbs (HA) and conjunctions (KOT) have quite high
precision but the tagger missed some of them. The open part of speech categories, such as nouns (FN), adjectives (MN) and verbs (IGE) have the lowest precision and recall. For the category numerals (SZN) the recall rate is maximum but the precision rate is very low. The result doesn’t say much about that category because the occurrence of this category is minimal.

6.3 Testing on the fairy tale

- The total number of tokens is 2110.
- The total number of correctly PoS tagged tokens retained by the tagger is 1793.
- The total number of tokens which have correct and fully complete PoS and subtags is 1715.
- The total number of tokens which have correct PoS and subtags but are missing some subtags is 1761.
- The total number of tokens without foreign names (Suryakanta, Balapandita and Razakosa) is 1999.

Thus, precision can be calculated for the PoS tagged fairy tale and for the PoS and subtagged fairy tale as shown in table h below. Here, in contrast to the poem, the version with PoS and subtags has the lowest accuracy no matter if tags with missing subtags are counted as wrong or as right tags (85% compared to 81.3% and 83.4%). Although the accuracy increases from 81.3% to 83.4%, when the missing tags for inflectional categories are excluded from consideration. In view of the fact that the foreign names Suryakanta, Balapandita and Razakosa (in total 111) were frequently wrongly annotated by the tagger the precision was also calculated without those words. When these foreign names are excluded from the total number of machine-tagged words, the accuracy increases considerably: from 85% to 86.6% for the PoS tagged version, and from 81.3% to 85.4% for the PoS and subtagged version excluding missing subtags as correct.

On the other hand, the accuracy is consistently higher for the PoS and subtagged version when only counting correct PoS tags, compared to the PoS tagged version.

Table H. Precision for fairy tale

<table>
<thead>
<tr>
<th>Fairy tale</th>
<th>correct tags in %</th>
<th>with PoS tags</th>
<th>with PoS and subtags</th>
</tr>
</thead>
<tbody>
<tr>
<td>only PoS tags</td>
<td>(1793/2110) 100 %</td>
<td>85%</td>
<td>(1829/2110) 100 %</td>
</tr>
<tr>
<td>excluding missing subtags as correct</td>
<td>- - -</td>
<td>(1715/2110) 100 %</td>
<td>81.3 %</td>
</tr>
<tr>
<td>including missing subtags as correct</td>
<td>- - -</td>
<td>(1761/2110) 100 %</td>
<td>83.4 %</td>
</tr>
<tr>
<td>without foreign names (and excluding missing subtags as correct)</td>
<td>(1731/1999) 100 %</td>
<td>86.6 %</td>
<td>(1708/1999) 100 %</td>
</tr>
<tr>
<td>without foreign names and including missing subtags as correct</td>
<td>- - -</td>
<td>(1754/1999) 100 %</td>
<td>87.7 %</td>
</tr>
<tr>
<td>without foreign names and only PoS tags</td>
<td>- - -</td>
<td>(1805/1999) 100 %</td>
<td>90.3 %</td>
</tr>
</tbody>
</table>

The precision and recall for different PoS tags are shown in table i below. Similarly to table g in section 6.2, in the first column different PoS tags are listed. In the second column the precision with the categories to which the wrongly annotated words actually belong are given, while in the third column the recall is calculated and the categories which the tagger wrongly annotated and missed are listed.
<table>
<thead>
<tr>
<th>PoS tags</th>
<th>Precision correct_found/retrieved_total</th>
<th>Recall correct_found/intended_total</th>
</tr>
</thead>
<tbody>
<tr>
<td>INF (Infinitive)</td>
<td>1.0 14/14 ----</td>
<td>1.0 = 14/14 ----</td>
</tr>
<tr>
<td>IGE (Verb)</td>
<td>0.69 260/377 FN 89 MN 9 NM 8 HA 5 ISZ 5 KOT 1</td>
<td>0.86 260/302 FN 28 MN 8 KOT 4 ISZ 5 SZN 2</td>
</tr>
<tr>
<td>IK (Verbal Particle)</td>
<td>0.73 22/30 NM 8</td>
<td>0.88 22/25 NM 2 HA 1</td>
</tr>
<tr>
<td>FN (Noun)</td>
<td>0.85 453/534 IGE 28 MN 22 NM 12 HA 11 ISZ 5 KOT 2 SZN 1</td>
<td>0.77 453/585 IGE 89 MN 35 HA 7 NM 1</td>
</tr>
<tr>
<td>DET (Determiner)</td>
<td>1.0 148/148 ----</td>
<td>1.0 148/148 ----</td>
</tr>
<tr>
<td>MN (Adjective)</td>
<td>0.69 116/168 FN 35 IGE 8 HA 5 ISZ 2 NM 1 KOT 1</td>
<td>0.78 116/148 FN 22 IGE 9 HA 1</td>
</tr>
<tr>
<td>NM (Pronoun)</td>
<td>0.94 118/125 KSZ 3 HA 2 FN 1 IK 1</td>
<td>0.80 118/148 FN 12 IGE 8 IK 7 HA 1 MN 1</td>
</tr>
<tr>
<td>HA (Adverb)</td>
<td>0.84 109/129 FN 7 KSZ 5 KOT 5 MN 1 NM 1 IK 1</td>
<td>0.72 109/152 KOT 13 FN 11 NU 7 IGE 5 MN 5 NM 2</td>
</tr>
<tr>
<td>SZN (Numeral)</td>
<td>0.78 7/9 IGE 2</td>
<td>0.88 7/8 FN 1</td>
</tr>
<tr>
<td>NU (Postposition)</td>
<td>0.82 31/38 HA 7</td>
<td>1.0 31/31 ----</td>
</tr>
<tr>
<td>KOT (Conjunction)</td>
<td>0.91 178/195 HA 13 IGE 4</td>
<td>0.96 178/186 HA 3 FN 2 MN 2 IGE 1</td>
</tr>
<tr>
<td>ISZ (Interjection)</td>
<td>1.0 5/5 ----</td>
<td>0.20 5/25 NP 8 FN 5 IGE 5 MN 2</td>
</tr>
<tr>
<td>KSZ (Interrogative)</td>
<td>-- 0 ----</td>
<td>-- 0/8 HA 5 MN 3</td>
</tr>
</tbody>
</table>
The categories infinitive (INF) and determiner (DET) have maximum recall and precision, thus the tagger found these categories easiest. For the category postposition (NU) the tagger didn’t miss any, but it wrongly suggested that words from this category were adverbs (HA) seven of 38 times. Pronouns (NM) and nouns (FN) have tolerably high precision but the tagger missed 30 of 148 pronouns and 132 of 453 nouns. The tagger had the most difficulties with adjectives (MN) and verbs (IGE) which were often incorrectly annotated and often missed. These results resemble the results of the precision and recall of PoS tags in the poem, where the open categories seem to be the most difficult ones for the tagger to correctly annotate.

### 6.4 Evaluation

Testing on the training set, i.e. using the same corpus for training and testing, gave the best result (98.6% and 98.8%), as would be predicted. Due to the fact that the tagger learned rules on the same corpus as the test corpus, the outcome of testing is much better than it is for the other types of test texts. The results don’t give a valid statement about the performance of the system, but tells how good or bad the rules the system derived from the training set are. These results mean that the tagger couldn’t correctly or completely annotate approximately one in every hundred words. One reason for this is the great number of homographs in Hungarian, which according to Pajzs’ (1996) examination, averages more than 30% of the running words in a corpus consisting of 200 000 running words. Thus, the tagger based on the lexical and contextual rules managed to correctly annotate most of the homographs.

In order to get a picture of the taggers performance, the tagger was tested on two different samples (the poem and the fairy tale) other than the training set. The results are lower than the result obtained from training and testing on the same text. In general, the tagger succeeded better in annotating the poem (85.9%-92.2%) than the fairy tale (81.3%-90.3%) with parts of speech with or without subtags. On the other hand, the poem consists of fewer tokens than the fairy tail, hence its result is not as reliable as the result of the fairy tail.

On average, putting together the results from both subcorpora consisting of 2507 words, we get 85.12% correct results for only part of speech tags, 82.45% for part of speech tags with correct and complete subtags, 84.44% for part of speech tags with correct but not necessarily complete subtags, and 87.55% for part of speech tags without regarding the correctness of the subtag. These results are not very promising when compared with Brill’s results of English test corpora which have an accuracy of 96.5% trained on 88200 words (Brill, 1995a). The difference in accuracy might depend on i) the type of the training corpus, ii) the types and the size of the test corpus, and iii) the type of language structure, such as morphology and syntax.

The corpus which was used to train the tagger on Hungarian consisted of only one text, a fiction with ‘inventive’ language, while Brill used a training corpus consisting of several types of texts (Brill, 1995a). Also, there is a difference between the types and the sizes of the test corpora. In this work very small samples, which greatly differ in type from the training corpus, have been used, while Brill’s test corpus consisted of different types of texts (Brill, 1995a). Nevertheless, the most significant difference between the results lies in the type of the language structure. Hungarian, as was described in section 4 has free word order and more complicated morphology than English has. I argue that the low tagging accuracy for Hungarian mostly depends on the fact that the templates of the learner modules of the tagger, especially the templates of the lexical learner module, are predefined in a way that they include strong language specific information which does not fit for Hungarian and other agglutinative languages. The predefined templates are principally based on the structure of English and perhaps other ‘big’ European languages,
even if Brill maintains that his system is language independent. Those lexical templates whose triggers depend on the affixes of a word look at only the first or last four characters of a word. In other words, defining that a lexical trigger is ‘delete/add the suffix x where |x| ≤ 4’ is to assert that it is only important to look at the last or first four letters in a word which is often not enough for correct annotation in Hungarian. For example, the word *siessu2nk* ‘hurry up:1PL’ was annotated by the tagger as IGE_t1, i.e. as a verb in present indicative first person plural with indefinite object. The correct annotation should be IGE_Pt1, i.e. as a verb in imperative (P) first person plural with indefinite object. Because the tagger was only looking at the last four characters *-u2nk*, it missed the necessary information about the imperative *-s*\(^{19}\). Another example concerns derivational suffixes giving important information about the PoS tag because they often change the category of the word. They follow the stem of the word and may be followed by different inflectional suffixes, see also section 4.2.9.2. For example, the word *al1rt:atlansalg:alt* ‘harm:less;Deadjectival_noun:ACC’ could be wrongly annotated by the tagger because information about the two derivational suffixes is missed if the word *al1natalsalg* doesn’t exist in the lexicon. Thus, if the tagger had looked at more than four characters it would have been possible to reduce the total number of words in the lexicon and the tagger would have been able to create better and more efficient rules concerning the morphological structure of Hungarian words. This is especially true in the case of the corpora used in this work, since the encoding of accentuation of the vowels is done with extra characters (numbers) which reduces the effective length of the affixes. In the example above, *siessu2nk* (*siessünk*), at most three of the last letters are examined.

For Hungarian, the triggers of templates seem to be unsuccessful because of the Hungarian suffix structure of the open classes, such as the categories Noun, Verb and Adjective, as it was described in section 4.2. A possible solution could be to give the user the possibility to change the predefined language specific templates to more suitable ones for the particular language in some easy way. One could for instance give a template file as an input to the learner.

Furthermore, the contextual rules probably work more effectively for strong ‘word-order’ languages, such as English, than for Hungarian, though it is difficult to show without evaluating the system more thoroughly.

In order to know which categories the tagger in general failed to identify, the precision and recall were calculated for each part of speech category of the poem and fairy tale together in a similar way as it was calculated for each type of text in sections 6.2 and 6.3. In table \(j\) the results are given.

\(^{19}\) Note that in imperative/subjunctive mood the marker is -j, but here the -j changes to -ss because the root of the word ends in -t (*siet* ‘hurry’), see also section 4.2.6.
<table>
<thead>
<tr>
<th>PoS tags</th>
<th>Precision</th>
<th>Recall</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>correct_found/retrieved_total</td>
<td>correct_found/intended_total</td>
</tr>
<tr>
<td>INF (Infinitive)</td>
<td>0.90 27/30</td>
<td>FN 1 IGE 1 MN 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.96 27/28 FN 1</td>
</tr>
<tr>
<td>IGE (Verb)</td>
<td>0.70 302/429</td>
<td>FN 93 MN 14 NM 8 HA 5 ISZ 5 KOT 1 INF 1 NU 1</td>
</tr>
<tr>
<td></td>
<td>0.83 302/360</td>
<td>FN 38 MN 12 KOT 4 SZN 3 INF 1</td>
</tr>
<tr>
<td>IK (Verbal Particle)</td>
<td>0.74 26/35</td>
<td>NM 9</td>
</tr>
<tr>
<td></td>
<td>0.84 26/31</td>
<td>NM 3 HA 2</td>
</tr>
<tr>
<td>FN (Noun)</td>
<td>0.83 523/627</td>
<td>IGE 38 MN 33 NM 13 HA 11 ISZ 5 KOT 2 INF 1 SZN 1</td>
</tr>
<tr>
<td></td>
<td>0.78 523/666</td>
<td>IGE 93 MN 41 HA 7 NM 1 INF 1</td>
</tr>
<tr>
<td>DET (Determiner)</td>
<td>1.0 173/173</td>
<td>----</td>
</tr>
<tr>
<td></td>
<td>1.0 173/173</td>
<td>----</td>
</tr>
<tr>
<td>MN (Adjective)</td>
<td>0.70 151/214</td>
<td>FN 41 IGE 12 HA 6 ISZ 2 NM 1 KOT 1</td>
</tr>
<tr>
<td></td>
<td>0.75 151/200</td>
<td>FN 33 IGE 14 HA 1 INF 1</td>
</tr>
<tr>
<td>NM (Pronoun)</td>
<td>0.94 132/140</td>
<td>KSZ 3 HA 2 IK 2 FN 1</td>
</tr>
<tr>
<td></td>
<td>0.80 132/164</td>
<td>FN 13 IK 9 IGE 8 HA 1 MN 1</td>
</tr>
<tr>
<td>HA (Adverb)</td>
<td>0.85 137/160</td>
<td>FN 7 KOT 6 KSZ 6 IK 2 MN 1 NM 1</td>
</tr>
<tr>
<td></td>
<td>0.74 137/184</td>
<td>KOT 16 FN 11 NU 7 MN 6 IGE 5 NM 2</td>
</tr>
<tr>
<td>SZN (Numeral)</td>
<td>0.73 8/11</td>
<td>IGE 3</td>
</tr>
<tr>
<td></td>
<td>0.89 8/9</td>
<td>FN 1</td>
</tr>
<tr>
<td>NU (Postposition)</td>
<td>0.83 34/41</td>
<td>HA 7</td>
</tr>
<tr>
<td></td>
<td>0.97 34/35</td>
<td>IGE 1</td>
</tr>
<tr>
<td>KOT (Conjunction)</td>
<td>0.91 200/220</td>
<td>HA 16 IGE 4</td>
</tr>
<tr>
<td></td>
<td>0.96 200/209</td>
<td>HA 4 FN 2 MN 2 IGE 1</td>
</tr>
<tr>
<td>ISZ (Interjection)</td>
<td>1.0 5/5</td>
<td>----</td>
</tr>
<tr>
<td></td>
<td>0.20 5/25</td>
<td>NNP 8 FN 5 IGE 5 MN 2</td>
</tr>
<tr>
<td>KSZ (Interrogative)</td>
<td>-- 0</td>
<td>----</td>
</tr>
<tr>
<td></td>
<td>-- 0/9</td>
<td>HA 6 NM 3</td>
</tr>
</tbody>
</table>
The tagger managed to find all determiners [DET] and didn’t give any wrong alternatives either which is in many ways astonishing because the definite article az ‘the’ may be homonymous with the demonstrative pronoun az ‘that’ and the indefinite article egy ‘a/an’ has the same form as the numeral ‘one’.

Interjections [ISZ] also have maximal precision, i.e. all words which have been annotated as interjections are correct but the tagger missed 20 of 25 interjections, hence the low recall (20%). 8 of the 25 interjections have been tagged with the default tag NNP meaning that the word begins with a capital letter and is unknown.

Most of the words annotated with the tag pronoun [NM] are correct, only 8 of 140 were wrong but the tagger missed some of the pronouns and annotated them mainly as nouns [FN], verbal particles [IK] or verbs [IGE]. The mistakes depend on the fact that pronouns take personal endings in a similar way as nouns and verbs do. When the tagger looks for the word endings (the last four characters in a word) it finds the same endings for pronouns, nouns and verbs. The mix-up of the verbal particles may be the result of the fact that the interrogative pronoun ki ‘who’ has the same form as the verbal particle ‘out’.

Conjunctions [KOT] and infinitives [INF] both have high recall (96%) and precision (91% and 90%, respectively). Adverbs are often tagged as conjunctions, perhaps because both are found in the same position in a sentence. Furthermore, conjunctions were confused with verbs four times because of the homographs vagy and mert. Vagy is either the conjunction ‘or’, or the second person singular present form of the verb van ‘be’. Mert may belong to the class of conjunctions with the meaning ‘because’ or to the verb class with the form of past tense third person singular, meaning ‘dare’. These conjunctions can easily be detected because they are often the first word in a sentence or a clause.

For the category adverb [HA] the tagger missed 47 of 184 adverbs and mostly annotated these words as conjunctions [KOT] or nouns [FN]. This result is not surprising because adverbs may appear in the same position as conjunctions, and adverbs and nouns may take the same personal endings, as was described in section 4.2. Concerning precision, the tagger correctly found 137 adverbs but wrongly annotated 23 words belonging mainly to the categories noun [FN], conjunctions [KOT], and interrogatives [KSZ].

Postpositions [NU] have high recall (97%) meaning that the tagger missed only one. On the other hand this class has very low precision because the tagger annotated seven adverbs as postpositions. The difference between adverbs and postpositions is, among other things, that the later category doesn’t take personal endings while adverbs do. Note also that many adverbs have the same stem as postpositions (see section 4.2.8 for more detail).

The open classes, such as nouns [FN], verbs [IGE] and adjectives [MN] are often wrongly annotated and confused by the tagger because of their complicated form, their similar morphological structure and the fairly free word order in Hungarian. Many words belonging to these classes are homographs, especially nouns and verbs, and verbs and adjectives (or participles). For that reason, these categories have both low precision and recall.

Among the closed classes, verbal particles [IK] and numerals [SZN] have the lowest precision (74% and 73%, respectively) - the tagger often gives these categories an incorrect analysis - but have a higher recall (84% and 89%, respectively), i.e. the tagger finds many of them. This is because of the fact that the words belonging to these categories constitute a countable set, many of them being in the lexicon but often ambiguous.

To sum up the results, the tagger has the most difficulties with categories belonging to the open classes because of their morphological structure and homonymy, while grammatical categories are easier to detect and correctly annotate. Complicated and highly developed morphological structure and fairly free word order, i.e. making positional
relationships less important, lead to lower accuracy when using Brill’s tagger with Hungarian.
7 Directions for future research

Firstly, it has to be pointed out that the results described in section 6 are based on a very small test corpus consisting of approximately 2500 running words. For a better evaluation of the ability of the tagger to annotate Hungarian texts based on the learned lexical and contextual rules, it would be necessary to test the tagger on a large corpus with different types of texts, including fiction, poetry, non-fiction, articles from different newspapers, trade journals, etc. Note that for effective evaluation it is important to have a correctly annotated test corpus, where the tagset and the annotation principle for that corpus are in accordance with the training corpus. The correctly annotated test corpus and the (by the tagger) reannotated test corpus can then be automatically compared.

It would be interesting to evaluate the components of the system for Hungarian by running only the lexical rules on the test corpus, and compare its accuracy to the accuracy when both sets of rules are applied, i.e. to see how much the contextual rules improve the result.

For a higher accuracy it would be necessary to train the tagger on a larger corpus with different types of texts or even on several corpora because the likelihood of higher accuracy increases with the size of the training corpus. Unfortunately, there is a trade-off between accuracy and training time; training on a large corpus can take a lot of time and memory, but on the other hand it gives higher accuracy.

It seems reasonable that before lexical training, the default tags NN and NNP should be changed in the program to the most common tag in the Hungarian tagset, e.g. FN (noun), which was not done for the training on the Hungarian corpus. On the other hand, the outcome of modifications like this is hard to predict since the rule generating process might be very dynamic.

Furthermore, it would be advantageous to create a very large dictionary of the type Word Tag, Tag, ... Tag, where the first tag is the most frequent tag for that word, listing all possible tags for each word. By using this lexicon, accuracy would be improved in two ways. First, the number of unknown words, i.e. the number of words not being in the training corpus would be reduced, though no matter how much text the tagger looks at there will always be a number of words that appear only a few times, according to Zipf’s law (frequency is roughly proportional to inverse rank). Secondly, the large dictionary would give more accurate knowledge about the set of possible part of speech tags for a particular word. For example, the template of the type ‘Change the most likely tag from X to Y, if...’, the template would only change tag X to tag Y, if tag Y exists with a particular word in the training corpus. Thus, a large dictionary would reduce the errors of the annotation by applying better rules and increase the speed of the contextual learning.

A limitation of the tagger lies in the fact that it has difficulty handling idioms and lexicalised phrases because there are restrictions on the length of the segment - only one word long segments can be marked. One possibility could be to construct a tagset with new tags for idioms/lexicalised phrases, where the first part of the tag is the name of the idiom/lexicalised phrase and the second part a number which indicates the position of the element in the idiom/lexicalised phrase.

Also, if would be desirable if the system could add the lemma of a certain word to its tag, since the lemma is often vital for further investigations. The lemma could for example stand in the first position of the compound tag. This would probably demand an even larger training corpus. How this notation would affect the learning process and the accuracy of the tagger is difficult to predict, but would be interesting to try. The lexical learning process would be even more time and memory consuming than it already is, hence this approach might not be feasible. Brill’s tagger already requires much memory.

---

20 The default tag NNP occurred eight times in the annotation of the test corpus though the tag does not exist in the Hungarian tagset.
and processing time, especially for the lexical rule learner; the lexical learning process based on 452 tags (PoS and subtag combinations) took one week.

The above mentioned suggestions for improvement are applicable by a future conscientious user. Below, some recommendations concerning the improvement of the system follow.

To get better accuracy it would be worthwhile attempting to change the predefined transformation templates so that they would work better for Hungarian and for other languages with a different morphological structure than English. The templates which limit the accuracy, in my opinion, are the lexical types, namely deleting/adding the suffix/prefix x, where $|x| \leq 4$ and the current word has prefix/suffix x, where $|x| \leq 4$ (see also section 3.2, template 2-7 and 10-15). It should be possible to set the maximum length of x by the user before training. For Hungarian, for example, it would be better if x has been set to 6 or 8, i.e. if the system could look at the first/last six or eight characters. Often it is not enough to look at the first/last four characters because the system loses information about other prefixes/suffixes which could give relevant information about a specific part of speech category or a relevant inflectional category, see also section 6.4.

An even better improvement would be that the user could supply his/her own template file to the learner/tagger. In today’s version of the tagger it is difficult to change the predefined transformation templates.
8 Conclusion

This work has presented Eric Brill’s rule-based PoS tagger which automatically acquires rules from a training corpus, based on transformation-based error driven learning. The tagger has been trained on a Hungarian corpus, which is composed of Orwell’s 1984 consisting of 99860 tokens including punctuation marks. The tagset of the training corpus consists of 452 PoS tags including inflectional properties of which 31 denote different parts of speech. The training was done two times, once for only part of speech tags and once for part of speech and ‘subtags’ denoting inflectional properties. New test texts with approximately 2500 words as well as the training corpus have been tested on the tagger. Precision was calculated for all test texts, and recall and precision for specific part of speech tags. The results presented in this work show that the accuracy of the test texts was 85.12% for only part of speech tags, 82.45% for part of speech tags with correct and complete subtags, and 84.44% for part of speech tags with correct but not necessarily complete subtags. The best annotation, 87.5% has been attained by tagging a text with PoS and subtags although only considering the correctness of the PoS tag (i.e. without regarding the subtag). The tagger had most difficulties with categories belonging to the open classes because of their complicated morphological structure while closed-class categories were easier to detect and correctly annotate.

It was shown, that it is probably more difficult to derive information from words which are annotated with only PoS tags, than from words whose tags include information about the inflectional categories.

It was suggested that to be able to more accurately evaluate Brill’s tagger on Hungarian it is necessary to test the tagger on a larger corpus and also to evaluate different components of the system. For higher accuracy it would be necessary to use a larger corpus which consists of different types of texts, to change the default tags to the most common tag in the Hungarian tagset, and to create a large lexicon of the type ‘Word Tag1 Tag2 ... TagN’ for reducing the number of unknown words.

Furthermore, it was shown that many of the predefined transformation templates of the two learning modules are language specific with the current choice of parameters. They are based on the language structure of English and perhaps other Germanic languages. They do not suit agglutinative languages, such as Hungarian because of their complicated morphological structure and fairly free word order. It would be necessary to construct a flexible solution giving the user the possibility to change the predefined transformation templates, or at least their parameters, in a simple way to better more expressive ones for a particular language.

Finally, there are many advantages of the tagger compared to other tagging systems, as Brill (1992) declares. Brill’s rule-based PoS tagger automatically learns its rules based on a comparably small correctly annotated corpus. The learners give a relatively small set of ‘meaningful’ rules. Furthermore, the tagset and/or corpus genre is easy to change to another. Also, it is easy to find and implement improvements to the tagger.
References


Appendix A: The Hungarian tagset

Part of speech Tags

- DET: Determiner
- FN: Noun
- NM: Pronoun
- KSZ: Interrogative
- MN: Adjective
- SZN: Numeral
- IGE: Verb
- INF: Infinitive
- IK: Verbal Particle
- HA: Adverb
- NU: Postposition
- KOT: Conjunction
- ISZ: Interjection
- MSZ: Sentence adverb
- MOD: Modifier
- USZ: Foreign word

Noun [FN]

Number

- PL: Plural  <k, ak, ek, ok, ök>

<table>
<thead>
<tr>
<th>Num/Per</th>
<th>Tag</th>
<th>Allomorph</th>
</tr>
</thead>
<tbody>
<tr>
<td>Singular 1</td>
<td>PSe1</td>
<td>&lt;m, am, em, om, öm&gt;</td>
</tr>
<tr>
<td>2</td>
<td>PSe2</td>
<td>&lt;d, ad, od, ed, öd&gt;</td>
</tr>
<tr>
<td>3</td>
<td>PSe3</td>
<td>&lt;a, e, ja, je, á, é, já, jé&gt;</td>
</tr>
<tr>
<td>Plural 1</td>
<td>PST1</td>
<td>&lt;nk, unk, ünk&gt;</td>
</tr>
<tr>
<td>2</td>
<td>PST2</td>
<td>&lt;tok, tek, tök, atok, etek, ötök&gt;</td>
</tr>
<tr>
<td>3</td>
<td>PST3</td>
<td>&lt;uk, ük, juk, jük&gt;</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Num/Per</th>
<th>Tag</th>
<th>Allomorph</th>
</tr>
</thead>
<tbody>
<tr>
<td>Singular 1</td>
<td>PST1i</td>
<td>&lt;im, aim, eim&gt;</td>
</tr>
<tr>
<td>2</td>
<td>PST2i</td>
<td>&lt;id, aid, eid&gt;</td>
</tr>
<tr>
<td>3</td>
<td>PST3i</td>
<td>&lt;i, ai, jai, jei&gt;</td>
</tr>
<tr>
<td>Plural 1</td>
<td>PST1ii</td>
<td>&lt;ink, aink, eink, jaink, jeink&gt;</td>
</tr>
<tr>
<td>2</td>
<td>PST2ii</td>
<td>&lt;itok, itek, jaitok, jeltek&gt;</td>
</tr>
<tr>
<td>3</td>
<td>PST3ii</td>
<td>&lt;ik, aik, eik, jaik, jeik&gt;</td>
</tr>
</tbody>
</table>

POS: Possessive  <é, éi>
<table>
<thead>
<tr>
<th>Tag</th>
<th>Name</th>
<th>Allomorph</th>
</tr>
</thead>
<tbody>
<tr>
<td>NO</td>
<td>Nominative</td>
<td>&lt;Ø&gt;</td>
</tr>
<tr>
<td>AC</td>
<td>Accusative</td>
<td>&lt;t, at, et, ot, öt&gt;</td>
</tr>
<tr>
<td>DA</td>
<td>Dative-genitive</td>
<td>&lt;nak, nek&gt;</td>
</tr>
<tr>
<td>IL</td>
<td>Illative</td>
<td>&lt;ba, be&gt;</td>
</tr>
<tr>
<td>IN</td>
<td>Inessive</td>
<td>&lt;ban, ben&gt;</td>
</tr>
<tr>
<td>EL</td>
<td>Elative</td>
<td>&lt;ból, bÓl&gt;</td>
</tr>
<tr>
<td>AL</td>
<td>Allative</td>
<td>&lt;hoz, hez, höz&gt;</td>
</tr>
<tr>
<td>AD</td>
<td>Adessive</td>
<td>&lt;nál, nél&gt;</td>
</tr>
<tr>
<td>AB</td>
<td>Ablative</td>
<td>&lt;tól, tÓl&gt;</td>
</tr>
<tr>
<td>SU</td>
<td>Sublative</td>
<td>&lt;ra, re&gt;</td>
</tr>
<tr>
<td>SU</td>
<td>Superessive</td>
<td>&lt;n, on, en, ön&gt;</td>
</tr>
<tr>
<td>DE</td>
<td>Delative</td>
<td>&lt;ról, rÓl&gt;</td>
</tr>
<tr>
<td>TE</td>
<td>Terminative</td>
<td>&lt;ig&gt;</td>
</tr>
<tr>
<td>CA</td>
<td>Causal-final</td>
<td>&lt;ért&gt;</td>
</tr>
<tr>
<td>IN</td>
<td>Instrumental</td>
<td>&lt;val, vel, al, el, bal, bel, etc.&gt;</td>
</tr>
<tr>
<td>FA</td>
<td>Translative</td>
<td>&lt;vá, vé, á, é, bá, bé, etc.&gt;</td>
</tr>
<tr>
<td>FO</td>
<td>Formal</td>
<td>&lt;ként, képp, képpen&gt;</td>
</tr>
<tr>
<td>SO</td>
<td>Essive-modal</td>
<td>&lt;stul, stül&gt;</td>
</tr>
<tr>
<td>TE</td>
<td>Temporal</td>
<td>&lt;kor&gt;</td>
</tr>
</tbody>
</table>

**Verb [IGE]**

**Present indicative, indefinite object**

<table>
<thead>
<tr>
<th>Num/Per</th>
<th>Tag</th>
<th>Allomorph</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>e1</td>
<td>&lt;ok, ek, ök&gt;</td>
</tr>
<tr>
<td>Singular</td>
<td>2</td>
<td>&lt;sz, asz, esz, ol, el, öl&gt;</td>
</tr>
<tr>
<td>3</td>
<td>e3</td>
<td>&lt;Ø&gt;</td>
</tr>
<tr>
<td>Plural</td>
<td>1</td>
<td>&lt;unk, ünk&gt;</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>&lt;tok, tek, tök, otok, etek, ötök&gt;</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>&lt;nak, nek, anak, enek&gt;</td>
</tr>
</tbody>
</table>
### Present indicative, definite object

<table>
<thead>
<tr>
<th>Num/Per</th>
<th>Tag</th>
<th>Allomorph</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Te 1</td>
<td>&lt;om, em, öm&gt;</td>
</tr>
<tr>
<td></td>
<td>Te 2</td>
<td>&lt;od, ed, öd&gt;</td>
</tr>
<tr>
<td></td>
<td>Te 3</td>
<td>&lt;ja, i&gt;</td>
</tr>
<tr>
<td>1</td>
<td>Tt 1</td>
<td>&lt;juk, jük, uk, ük&gt;</td>
</tr>
<tr>
<td></td>
<td>Tt 2</td>
<td>&lt;játok, itek&gt;</td>
</tr>
<tr>
<td></td>
<td>Tt 3</td>
<td>&lt;ják, ik&gt;</td>
</tr>
</tbody>
</table>

### Past, indefinite object

<table>
<thead>
<tr>
<th>Num/Per</th>
<th>Tag</th>
<th>Allomorph</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Me1</td>
<td>&lt;tam, tem, ttam, ttem, ottam, ettem, öttem&gt;</td>
</tr>
<tr>
<td></td>
<td>Me2</td>
<td>&lt;tal, tel, ttal, ttél, ottál, ettél, öttél&gt;</td>
</tr>
<tr>
<td></td>
<td>Me3</td>
<td>&lt;ta, te, tta, tte, otta, ette, ötte&gt;</td>
</tr>
<tr>
<td>1</td>
<td>Mt1</td>
<td>&lt;tunk, tünk, ttunk, ttünk, ottunk, ettünk, öttünk&gt;</td>
</tr>
<tr>
<td></td>
<td>Mt2</td>
<td>&lt;tatok, tetek, ttatok, ttatek, ottatok, ettetek, öttetek&gt;</td>
</tr>
<tr>
<td></td>
<td>Mt3</td>
<td>&lt;tak, tek, tták, tték, ottak, ettek, ötték&gt;</td>
</tr>
</tbody>
</table>

### Past, definite object

<table>
<thead>
<tr>
<th>Num/Per</th>
<th>Tag</th>
<th>Allomorph</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TMe1</td>
<td>&lt;tam, tem, ttam, ttem, ottam, ettem, öttem&gt;</td>
</tr>
<tr>
<td></td>
<td>TMe2</td>
<td>&lt;tad, ted, ttad, tted, ottad, etted, ötted&gt;</td>
</tr>
<tr>
<td></td>
<td>TMe3</td>
<td>&lt;ta, te, tta, tte, otta, ette, ötte&gt;</td>
</tr>
<tr>
<td>1</td>
<td>TMt1</td>
<td>&lt;tuk, tük, ttuk, ttük, ottuk, ettük, öttük&gt;</td>
</tr>
<tr>
<td></td>
<td>TMt2</td>
<td>&lt;tátok, tétek, ttátok, ttétek, ottátok, ettétek, öttétek&gt;</td>
</tr>
<tr>
<td></td>
<td>TMt3</td>
<td>&lt;ták, ték, tták, tték, ották, ettek, ötték&gt;</td>
</tr>
</tbody>
</table>

### Subjunctive/imperative, indefinite object

<table>
<thead>
<tr>
<th>Num/Per</th>
<th>Tag</th>
<th>Allomorph</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Pe1</td>
<td>&lt;jak, jek, ak, ek&gt;</td>
</tr>
<tr>
<td></td>
<td>Pe2</td>
<td>&lt;jál, jél, ál, él, j&gt;</td>
</tr>
<tr>
<td></td>
<td>Pe3</td>
<td>&lt;jon, jen, jön, on, en, ön, ék&gt;</td>
</tr>
<tr>
<td>1</td>
<td>Pt1</td>
<td>&lt;junk, jünk, unk, ünk&gt;</td>
</tr>
<tr>
<td></td>
<td>Pt2</td>
<td>&lt;jatok, jetek, atok, etek&gt;</td>
</tr>
<tr>
<td></td>
<td>Pt3</td>
<td>&lt;janak, jenek, anak, enek&gt;</td>
</tr>
</tbody>
</table>

### Subjunctive/imperative, definite object

<table>
<thead>
<tr>
<th>Num/Per</th>
<th>Tag</th>
<th>Allomorph</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TPe1</td>
<td>&lt;jam, jem, am, em&gt;</td>
</tr>
<tr>
<td></td>
<td>TPe2</td>
<td>&lt;jad, jed, ad, ed&gt;</td>
</tr>
<tr>
<td></td>
<td>TPe3</td>
<td>&lt;ja, je, a, e&gt;</td>
</tr>
<tr>
<td>1</td>
<td>TPt1</td>
<td>&lt;juk, jük, uk, ük&gt;</td>
</tr>
</tbody>
</table>
### Conditional, indefinite object

<table>
<thead>
<tr>
<th>Num/Per</th>
<th>Tag</th>
<th>Allomorph</th>
</tr>
</thead>
<tbody>
<tr>
<td>Singular 1</td>
<td>Fe1</td>
<td>&lt;nék, anék, enék, nám, ném&gt;</td>
</tr>
<tr>
<td></td>
<td>Fe2</td>
<td>&lt;nál, nél, anál, enél&gt;</td>
</tr>
<tr>
<td></td>
<td>Fe3</td>
<td>&lt;na, ne, ana, ena, nék&gt;</td>
</tr>
<tr>
<td>Plural   1</td>
<td>Ft1</td>
<td>&lt;nánk, nénk, anánk, enénk&gt;</td>
</tr>
<tr>
<td></td>
<td>Ft2</td>
<td>&lt;nátok, nétek, anátok, enétek&gt;</td>
</tr>
<tr>
<td></td>
<td>Ft3</td>
<td>&lt;nák, nék, anák, enék&gt;</td>
</tr>
</tbody>
</table>

### Conditional, definite object

<table>
<thead>
<tr>
<th>Num/Per</th>
<th>Tag</th>
<th>Allomorph</th>
</tr>
</thead>
<tbody>
<tr>
<td>Singular 1</td>
<td>TFe1</td>
<td>&lt;nám, ném, anám, eném&gt;</td>
</tr>
<tr>
<td></td>
<td>TFe2</td>
<td>&lt;nád, néd, anád, enéd&gt;</td>
</tr>
<tr>
<td></td>
<td>TFe3</td>
<td>&lt;ná, né, aná, ené&gt;</td>
</tr>
<tr>
<td>Plural   1</td>
<td>TFt1</td>
<td>&lt;nánk, nénk, anánk, enénk&gt;</td>
</tr>
<tr>
<td></td>
<td>TFt2</td>
<td>&lt;nátok, nétek, anátok, enétek&gt;</td>
</tr>
<tr>
<td></td>
<td>TFt3</td>
<td>&lt;nák, nék, anák, enék&gt;</td>
</tr>
</tbody>
</table>
The object is already marked on the verb; the suffix signals that the subject is first person singular and the object is second person.

<table>
<thead>
<tr>
<th>Tense/Mood</th>
<th>Tag</th>
<th>Allomorph</th>
</tr>
</thead>
<tbody>
<tr>
<td>present, indicative</td>
<td>Ie1</td>
<td>&lt;lak, lek, alak, elek&gt;</td>
</tr>
<tr>
<td>past</td>
<td>IMe1</td>
<td>&lt;talak, telek, ttalak, ttelek, ottalak, öttelek, ettelek&gt;</td>
</tr>
<tr>
<td>subjunctive</td>
<td>IFe1</td>
<td>&lt;nálak, nélek, análak, enélek&gt;</td>
</tr>
<tr>
<td>conditional</td>
<td>IPe1</td>
<td>&lt;jalak, jelek, alak, elek&gt;</td>
</tr>
</tbody>
</table>

Inflected infinitive forms

<table>
<thead>
<tr>
<th>Num/Per</th>
<th>Tag</th>
<th>Allomorph</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>INRe1</td>
<td>&lt;nom, nem, nöm, anom, enem, önöm&gt;</td>
</tr>
<tr>
<td>2</td>
<td>INRe2</td>
<td>&lt;nod, ned, nöd, anod, ened, önöd&gt;</td>
</tr>
<tr>
<td>3</td>
<td>INRe3</td>
<td>&lt;nia, nie, ania, enie&gt;</td>
</tr>
<tr>
<td>1</td>
<td>INRt1</td>
<td>&lt;nunk, nünk, anunk, enünk&gt;</td>
</tr>
<tr>
<td>2</td>
<td>INRt2</td>
<td>&lt;notok, netek, nötök, anotok, enetek, önötök&gt;</td>
</tr>
<tr>
<td>3</td>
<td>INRt3</td>
<td>&lt;niuk, niük, aniuk, eniük, niök, niök&gt;</td>
</tr>
</tbody>
</table>
# Appendix B: Lexical rules

There are 18 types of lexical rules, which are instantiated from a set of transformation templates, see also section 3.2. The types with their form and explanation are given in the table below. (Note that rules starting with ‘f’ are only applied if the current tag matches the specified current tag, while other rules change the tagging regardless of the current tag.)

<table>
<thead>
<tr>
<th>Type</th>
<th>Form</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>addpref</td>
<td>Ltr, addpref X Tag Score</td>
<td>If adding the letters Ltr of length X to the beginning of a word results in a word, tag it as Tag.</td>
</tr>
<tr>
<td>addsuf</td>
<td>Ltr, addsuf X Tag Score</td>
<td>If adding the letters Ltr of length X to the end of a word results in a word, tag it as Tag.</td>
</tr>
<tr>
<td>char</td>
<td>Ltr char Tag Score</td>
<td>If the character Ltr appears anywhere in the current word, tag it as Tag.</td>
</tr>
<tr>
<td>deletepref</td>
<td>Ltr, deletepref X Tag Score</td>
<td>If deleting the letters Ltr of length X from the beginning of a word results in a word, tag it as Tag.</td>
</tr>
<tr>
<td>deletesuf</td>
<td>Ltr, deletesuf X Tag Score</td>
<td>If deleting the letters Ltr of length X from the end of a word results in a word, tag it as Tag.</td>
</tr>
<tr>
<td>goodleft</td>
<td>Word goodleft Tag Score</td>
<td>If the current word ever appears to the left of the Word, tag it as Tag.</td>
</tr>
<tr>
<td>goodright</td>
<td>Word goodright Tag Score</td>
<td>If the current word ever appears to the right of the Word, tag it as Tag.</td>
</tr>
<tr>
<td>haspref</td>
<td>Ltr, haspref X Tag Score</td>
<td>If a word has the prefix Ltr of length X, tag it as Tag.</td>
</tr>
<tr>
<td>hassuf</td>
<td>Ltr, hassuf X Tag Score</td>
<td>If a word has the suffix Ltr of length X, tag it as Tag.</td>
</tr>
<tr>
<td>faddpref</td>
<td>Tag1 Ltr, faddpref X Tag2 Score</td>
<td>If the current word has tag Tag1 and if adding the letters Ltr of the length X to the beginning of a word results in a word, then change Tag1 to Tag2.</td>
</tr>
<tr>
<td>faddsuf</td>
<td>Tag1 Ltr, faddsuf X Tag2 Score</td>
<td>If the current word has tag Tag1 and if adding the letters Ltr of length X to the end of a word results in a word, then change Tag1 to Tag2.</td>
</tr>
<tr>
<td>fchar</td>
<td>Tag1 Ltr fchar Tag2 Score</td>
<td>If the character Ltr appears anywhere in the current word, and that word is tagged with Tag1, then change Tag1 to Tag2.</td>
</tr>
<tr>
<td>fdeletepref</td>
<td>Tag1 Ltr, fdeletepref X Tag2 Score</td>
<td>If the current word has tag Tag1 and if deleting the letters Ltr of length X from the beginning of a word results in a word, then change Tag1 to Tag2.</td>
</tr>
<tr>
<td>fdeletesuf</td>
<td>Tag1 Ltr, fdeletesuf X Tag2 Score</td>
<td>If the current word has tag Tag1 and if deleting the letters Ltr of length X from the end of a word results in a word, then change Tag1 to Tag2.</td>
</tr>
</tbody>
</table>
If the current word ever appears to the left of the Word and the current word is tagged as Tag1, then change Tag1 to Tag2.

If the current word ever appears to the right of the Word and the current word is tagged as Tag1, then change Tag1 to Tag2.

If a word has the prefix Ltr, of length X and it is currently tagged as Tag1, then change Tag1 to Tag2.

If a word has the suffix Ltr, of length X and it is currently tagged as Tag1, then change Tag1 to Tag2.

For example, rule

\[ \text{kus hassuf 3 MN 11} \]

means that if the last three characters of the word are kus, annotate the word with tag MN (as an adjective).

The rule

\[ \text{FN enek fhassuf 4 MN 11} \]

means that if the word is tagged with FN (as a noun) and the last four characters of that word is suffix enek, change the tag FN (noun) to tag MN (adjective).

In both rules above, the number in the last column indicates the score which the rule in question achieved during the training process and they only serve as place-holders in tagging, and can be replaced with any character string.
Lexical rules for PoS tags

1 char FN 3770.37859264918
NN t fchar IGE 1096.7107042863
a goodright FN 430.588443407703
ni hassuf 2 INF 380
NN o fchar FN 299.653846153846
1k hassuf 2 IGE 198
tt hassuf 2 IGE 193.098531746032
3 hassuf 1 MN 178.569298245614
NN e fchar IGE 165
1 hassuf 1 MN 153.952380952381
es hassuf 2 MN 127.666666666667
NN a fchar FN 126.5
ta hassuf 2 IGE 108
os hassuf 2 MN 104.8
lt hassuf 2 IGE 92.9976190476191
NNP n fchar FN 91.666666666667
FN i fhassuf 1 MN 90.2818181818182
bb hassuf 2 MN 83.6899766899767
tak hassuf 3 IGE 78
va hassuf 2 HA 76.2777777777778
ve hassuf 2 HA 68
et hassuf 2 FN 66.666666666667
len hassuf 3 MN 58.5
FN nem fgoodright IGE 57.3766840664407
sen hassuf 3 MN 54
IGE az fgoodright FN 52.8033456739339
an deletesuf 2 MN 51
te hassuf 2 IGE 50.2833333333333
ik hassuf 2 IGE 47
t addsuf 1 FN 44.9822169059011
l hassuf 1 FN 45.6698841698842
ni addsuf 2 IGE 40.4343434343434
lan hassuf 3 MN 39.666666666667
am haspref 2 MN 39.216393442623
NNP e fchar FN 35.8571428571429
IGE a fgoodright FN 33.8825396825397
nk hassuf 2 IGE 32
nu2l hassuf 4 MN 31
IGE volt fgoodleft FN 29.416666666667
nia hassuf 3 IGE 29
tek hassuf 3 IGE 28
ei hassuf 2 FN 27
3en hassuf 3 MN 25.3333333333333
IGE n fhassuf 1 FN 23
FN nagyon fgoodright MN 23.5714285714286
NNP S-T-A-R-T fgoodright NM 23
leg hassuf 3 HA 22.666666666667
FN tt fhassuf 2 MN 22.166666666666667
IGE egy fgoodright FN 21.4137931034483
jon hassuf 3 IGE 21
nul hassuf 3 MN 21
IGE it fhassuf 2 FN 20
anak hassuf 4 IGE 19
FN an faddsuf 2 MN 18.2
aik hassuf 3 FN 5
j deletesuf 1 IGE 5
ssa hassuf 3 IGE 5
lsz hassuf 3 IGE 5
nelm hassuf 4 IGE 5
znak hassuf 4 IGE 5
ju2k hassuf 4 IGE 5
jo2n hassuf 4 IGE 5
ol addsuf 2 IGE 5
INF i fdeletesuf 1 MN 5
FN bban fhassuf 4 MN 5
prol haspref 4 MN 5
ebb addsuf 3 MN 5
ha hassuf 2 HA 5
NM tt fhassuf 2 HA 5
zor hassuf 3 HA 5
lo3l hassuf 4 HA 5
Mag haspref 3 NM 5
FN tt fhassuf 2 MN 4.78571428571429
FN jo2 fhaspref 3 MN 4.6
IGE be fhassuf 2 FN 4.5
FN Eg fhaspref 2 HA 4.5
ls deletesuf 2 MN 4.12121212121212
mi deletepref 2 HA 4.01234567901235
NN i fchar FN 4
NN u fchar FN 4
ai deletesuf 2 FN 4
IGE ba fhassuf 2 FN 4
i01 hassuf 3 FN 4
ntja hassuf 4 FN 4
IGE o2sz fhaspref 4 FN 4
IGE roll faddsuf 4 FN 4
IGE volt fgoodleft FN 4
IGE voltak fgoodleft FN 4
IGE ulj fgoodright FN 4
dsz hassuf 3 IGE 4
nja hassuf 3 IGE 4
ant hassuf 3 IGE 4
tsa hassuf 3 IGE 4
zza hassuf 3 IGE 4
dja hassuf 3 IGE 4
lgja hassuf 4 IGE 4
heti hassuf 4 IGE 4
o3 addsuf 2 IGE 4
IGE mi fhassuf 2 MN 4
tan hassuf 3 MN 4
1ak hassuf 3 MN 4
ols hassuf 3 MN 4
snek deletesuf 4 MN 4
ajta hassuf 4 MN 4
he1r hassuf 4 MN 4
r01g haspref 4 MN 4
dolog goodleft MN 4
IGE egy fgoodright MN 4
nan hassuf 3 HA 4
FN sem fhassuf 3 HA 4
olta deletesuf 4 HA 4
ulgy hassuf 4 HA 4
jebb hassuf 4 HA 4
HA z fhassuf 1 NM 4
to3 haspref 3 NM 4
o2ve haspref 4 NM 4
egyi haspref 4 NM 4
benn haspref 4 NM 4
am addpref 2 NM 4
hogy addsuf 4 HA 4
Tiz haspref 3 SZN 4
lio1 hassuf 4 SZN 4
zalz hassuf 4 SZN 4
FN o2tv fhaspref 4 SZN 4
NN . fgoodleft SZN 4
csak deletesuf 4 KOT 4
-t deletesuf 2 USZ 4
Tim haspref 3 USZ 4
lelv addpref 4 NM 3.88235294117647
ki deletesuf 2 NM 3.60714285714286
FN 2lt fhassuf 3 IGE 3.5
FN dt fhassuf 2 MN 3.5
lyol hassuf 4 FN 3.33333333333333
etek deletesuf 4 FN 3.33333333333333
MN megs fhaspref 4 IGE 3.33333333333333
FN Az fhaspref 2 NM 3.33333333333333
alatt goodleft FN 3.32142857142857
FN mi faddsuf 2 KOT 3.10087719298246
IGE C fchar FN 3
NM j fchar FN 3
NNP a fchar FN 3
NNP i fchar FN 3
NM T fhaspref 1 FN 3
IGE pa fhaspref 2 FN 3
zel hassuf 3 FN 3
ink hassuf 3 FN 3
o3i hassuf 3 FN 3
jai hassuf 3 FN 3
IGE knak fhassuf 4 FN 3
IGE knek fhassuf 4 FN 3
IGE test fhaspref 4 FN 3
ls addsuf 2 FN 3
IGE ig faddsuf 2 FN 3
val addsuf 3 FN 3
o2n addsuf 3 FN 3
kkal addsuf 4 FN 3
IGE else3 fgoodright FN 3
IGE egelsz fgoodright FN 3
na hassuf 2 IGE 3
IGE 1t faddsuf 2 FN 4
FN ol fhassuf 2 IGE 3
jen deletesuf 3 IGE 3
rsz hassuf 3 IGE 3
bja hassuf 3 IGE 3
FN nom fhassuf 3 IGE 3
szel hassuf 4 IGE 3
sson hassuf 4 IGE 3
1tok hassuf 4 IGE 3
IGE kal faddu 3 FN 3
FN majd fgoodleft IGE 3
li hassuf 2 MN 3
Ro haspref 2 MN 3
uls hassuf 3 MN 3
lan hassuf 3 MN 3
FN sta fhasu 3 MN 3
ege haspref 3 MN 3
anok hassuf 4 MN 3
osra hassuf 4 MN 3
keny hassuf 4 MN 3
kete hassuf 4 MN 3
bbre hassuf 4 MN 3
2rke hassuf 4 MN 3
bnek hassuf 4 MN 3
FN nnek fhasu 4 MN 3
IGE snak fhasu 4 MN 3
IGE nne1 fhasu 4 MN 3
FN yire fhasu 4 MN 3
to2b haspref 4 MN 3
IGE 1n faddu 2 MN 3
MN ko2ru2l fgoodleft FN 3
pp hassuf 2 HA 3
hol hassuf 3 HA 3
nte hassuf 3 HA 3
ogy hassuf 3 HA 3
FN Mos fhasu 3 HA 3
ltal hassuf 4 HA 3
zo2r hassu 4 HA 3
yett hassu 4 HA 3
FN nben fhasu 4 HA 3
NM folytatta fgoodright HA 3
valahol goodright HA 3
FN gondolta fgoodright HA 3
MN 1ny fdeletesu 3 NM 3
mag deletepref 3 NM 3
na1 haspref 3 NM 3
nek haspref 3 NM 3
lyen deletesu 4 NM 3
roll haspref 4 NM 3
2 haspref 1 SZN 3
NM k fhasp 1 SZN 3
ann haspref 3 SZN 3
FN til fhasp 3 SZN 3
Ne1g haspref 4 SZN 3
FN nap fgoodleft SZN 3
FN tte fhasu 3 NU 3
HA telekelp fgoodleft DET 2.99532710280374
Lexical rules for PoS and subtags

o char MN 838.393967557783
t hassuf 1 IGE_Me3 659.101902858138
ni hassuf 2 INF 382.05
NN 1 fchar FN 363.165934065934
k hassuf 1 FN_PL 300.05
ta hassuf 2 IGE_TMe3 208.551231527094
lt hassuf 2 FN_PSe3_ACC 202.857142857143
te hassuf 2 IGE_TMe3 192.417741935484
l hassuf 1 FN_INS 181.007142857143
n hassuf 1 FN_INE 177.741666666667
tak hassuf 3 IGE_Mt3 130
lk hassuf 2 IGE_TMt3 128.25
IGE_Me3 a fgoodright FN_ACC 125.326767676768
ra hassuf 2 FN_SUB 113.05
lt addsuf 2 FN_PSe3 110.5
es hassuf 2 MN 99.6
bb hassuf 2 MN_FOK 96.477777777778
tek hassuf 3 IGE_Mt3 95.166666666667
nak hassuf 3 FN_DAT 94.1428571428571
g hassuf 1 FN 93.5434782608696
ln hassuf 2 FN_PSe3_SUP 80
va hassuf 2 HA 77.25
kat hassuf 3 FN_PL_ACC 77
ve hassuf 2 HA 76
ik hassuf 2 IGE_e3 74.4
et hassuf 2 FN_ACC 73.666666666667
re hassuf 2 FN_SUB 72.666666666667
ls hassuf 2 FN 69
len hassuf 3 MN 66.386666666667
NN i fchar MN 66.0147058823529
MN a fhassuf 1 FN_PSe3 65
ot hassuf 2 FN_ACC 58
st hassuf 2 FN_ACC 57
FN_INE en fdeletesuf 2 MN_ESS_MOD 57
an deletesuf 2 MN_ESS_MOD 52
eket hassuf 4 FN_PL_ACC 52
ba hassuf 2 FN_ILL 50
ni addsuf 2 IGE 49.5797101449275
nek hassuf 3 FN_DAT 48.75
FN i fhassuf 1 MN 48
nk hassuf 2 IGE_t1 48
m hassuf 1 FN 47.5
on hassuf 2 FN_SUP 46
it hassuf 2 FN_PSe3i_ACC 46
FN e fhassuf 1 FN_PSe3 45
FN_INS 11 fhassuf 2 FN_DEL 44
lan hassuf 3 MN 43
ja hassuf 2 IGE_Te3 42.3
bol l hassuf 4 FN_ELA 41
FN_INS 21 fhassuf 2 MN_ESS_MOD 39
r hassuf 1 FN 36.025641025641
3 hassuf 1 MN 36
jalk hassuf 4 IGE_Tt3 35.475
ot addsuf 2 FN 35
FN m fdeletesuf 1 FN_PSe1 4
vel haspref 3 NM_INS 4
ze1k hassuf 4 IGE_TPt3 4
FN_DEL j fchar FN_PSe3_DEL 4
benn haspref 4 NM_INE 4
ire hassuf 3 MN_SUB 4
FN_ALL a fgoodleft NM_ALL 4
dd hassuf 2 IGE_TPe2 4
1ik hassuf 3 FN_PSt3i 4
bnak hassuf 4 MN_FOK_DAT 4
bnek hassuf 4 MN_FOK_DAT 4
1tok hassuf 4 IGE_Tt2 4
nalk hassuf 4 IGE_Tt3 4
bbek hassuf 4 MN_FOK_PL 4
e1 deletesuf 2 FN_POS 3.95
MN kell fgoodleft IK 3.92857142857143
FN_ACC lt fhassuf 2 MN 3.6
MN t fdeletesuf 1 FN_ACC 4.73333333333333
va addsuf 2 IGE 3.55849802371542
SZN egelsz fgoodleft DET 3.55393851497289
rem hassuf 3 IGE_Tel 3.5
IGE_Me3 A fgoodright MN 3.25
vu2l hassuf 4 HA 3.07142857142857
MN sem fgoodleft FN 3.05263157894737
uls hassuf 3 MN 3
leti hassuf 4 MN 3
keny hassuf 4 MN 3
kete hassuf 4 MN 3
helr hassuf 4 MN 3
ista hassuf 4 MN 3
IGE_Me3 megv fhaspref 4 MN 3
ebb addsuf 3 MN 3
FN_PL volt fgoodleft MN 3
IGE_TPe3 c fchar FN 3
SZN p fchar FN 3
FN_PSe3_SUP - fchar FN 3
als deletesuf 3 FN 3
1tel hassuf 4 FN 3
lelk hassuf 4 FN 3
MN bo31 faddsuf 4 FN 3
MN alatt fgoodleft FN 3
IGE_e2 az fgoodright FN 3
FN_CAU z fchar IGE_Me3 3
IGE_Me3 mi fhaspref 2 HA 3
be1 hassuf 3 HA 3
1je hassuf 3 HA 3
ltal hassuf 4 HA 3
zo2r hassuf 4 HA 3
Most haspref 4 HA 3
egya haspref 4 HA 3
MN_ESS_MOD bizonyos fgoodleft HA 3
FN mindenki fgoodleft HA 3
FN_INS s fgoodright HA 3
FN_PSe3_SUB S-T-A-R-T fgoodright HA 3
FN_FOR S-T-A-R-T fgoodright HA 3
ult hassuf 3 FN_ACC 3
mben deletesuf 4 FN_PSel_INE 3
khez hassuf 4 FN_PL_ALL 3
FN_PL_INE mi fgoodright FN_PSt1_INE 3
mat deletesuf 3 FN_PSel_ACC 3
IGE_Me3_KSZ a fchar IGE_KSZ 3
ba1 hassuf 3 MN_FOK_FAC 3
SZN b fchar SZN_INE 3
a-e hassuf 3 IGE_TMe3_KSZ 3
ld deletesuf 2 IGE_TFe2 3
nkre hassuf 4 FN_PSt1_SUB 3
ikat hassuf 4 FN_KIEM_ACC 3
balr addsuf 4 KOT 2.99386503067485
Appendix C: Contextual rules

There are several types of contextual rules, which are instantiated from a set of transformation templates, see also section 3.3. The following types of rules are represented among the set of rules for Hungarian and are given in the table below with their form and explanation.

<table>
<thead>
<tr>
<th>Type</th>
<th>Form</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>CURWD</td>
<td>Tag1 Tag2 CURWD Word</td>
<td>Change Tag1 to Tag2 if the current word is Word.</td>
</tr>
<tr>
<td>LBIGRAM</td>
<td>Tag1 Tag2 LBIGRAM Word1 Word2</td>
<td>Change Tag1 to Tag2 if the left bigram are the words Word1 Word2 where current word is Word2.</td>
</tr>
<tr>
<td>NEXTBIGRAM</td>
<td>Tag1 Tag2 NEXTBIGRAM Tag3 Tag4</td>
<td>Change Tag1 to Tag2 if the following two tags are Tag3 and Tag4.</td>
</tr>
<tr>
<td>NEXTTAG</td>
<td>Tag1 Tag2 NEXTTAG Tag3</td>
<td>Change Tag1 to Tag2 if the next tag is Tag3.</td>
</tr>
<tr>
<td>NEXT1OR2TAG</td>
<td>Tag1 Tag2 NEXT1OR2TAG Tag3</td>
<td>Change Tag1 to Tag2 if any of the next two tags is Tag3.</td>
</tr>
<tr>
<td>NEXT1OR2OR3TAG</td>
<td>Tag1 Tag2 NEXT1OR2OR3TAG Tag3</td>
<td>Change Tag1 to Tag2 if any of the next three tags is Tag3.</td>
</tr>
<tr>
<td>NEXT2TAG</td>
<td>Tag1 Tag2 NEXT2TAG Tag3</td>
<td>Change Tag1 to Tag2 if the second following tag is Tag3.</td>
</tr>
<tr>
<td>NEXTWD</td>
<td>Tag1 Tag2 NEXTWD Word</td>
<td>Change Tag1 to Tag2 if the next word is Word.</td>
</tr>
<tr>
<td>NEXT1OR2WD</td>
<td>Tag1 Tag2 NEXT1OR2WD Word</td>
<td>Change Tag1 to Tag2 if any of the next two words is Word.</td>
</tr>
<tr>
<td>NEXT2WD</td>
<td>Tag1 Tag2 NEXT2WD Word</td>
<td>Change Tag1 to Tag2 if the second following word is Word.</td>
</tr>
<tr>
<td>PREVBIGRAM</td>
<td>Tag1 Tag2 PREVBIGRAM Tag3 Tag4</td>
<td>Change Tag1 to Tag2 if the previous two tags are Tag3 and Tag4.</td>
</tr>
<tr>
<td>PREVTAG</td>
<td>Tag1 Tag2 PREVTAG Tag3</td>
<td>Change Tag1 to Tag2 if the previous tag is Tag3.</td>
</tr>
<tr>
<td>PREV1OR2TAG</td>
<td>Tag1 Tag2 PREV1OR2TAG Tag3</td>
<td>Change Tag1 to Tag2 if any of the previous two tags is Tag3.</td>
</tr>
<tr>
<td>Tag Set</td>
<td>Description</td>
<td>Rule</td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
<td>------</td>
</tr>
<tr>
<td>PREV1OR2OR3TAG</td>
<td>Tag1 Tag2 PREV1OR2OR3TAG Tag3</td>
<td>Change Tag1 to Tag2 if any of the previous three tags is Tag3.</td>
</tr>
<tr>
<td>PREV2TAG</td>
<td>Tag1 Tag2 PREV2TAG Tag3</td>
<td>Change Tag1 to Tag2 if the second previous tag is Tag3.</td>
</tr>
<tr>
<td>PREVWWD</td>
<td>Tag1 Tag2 PREVWWD Word</td>
<td>Change Tag1 to Tag2 if the previous word is Word.</td>
</tr>
<tr>
<td>PREV1OR2WWD</td>
<td>Tag1 Tag2 PREV1OR2WWD Word</td>
<td>Change Tag1 to Tag2 if any of the previous two words is Word.</td>
</tr>
<tr>
<td>PREV2WWD</td>
<td>Tag1 Tag2 PREV2WWD Word</td>
<td>Change Tag1 to Tag2 if the second word before is Word.</td>
</tr>
<tr>
<td>RBIGRAM</td>
<td>Tag1 Tag2 RBIGRAM Word1 Word2</td>
<td>Change Tag1 to Tag2 if the right bigram is the words Word1 Word2 where current word is Word1.</td>
</tr>
<tr>
<td>SURROUNDTAG</td>
<td>Tag1 Tag2 SURROUNDTAG Tag3 Tag4</td>
<td>Change Tag1 to Tag2 if the previous tag is Tag3 and the next tag is Tag4.</td>
</tr>
<tr>
<td>WDAND2BFR</td>
<td>Tag1 Tag2 WDAND2BFR Word1 Word2</td>
<td>Change Tag1 to Tag2 if the word is Word2 and the second word before is Word1.</td>
</tr>
<tr>
<td>WDAND2TAGAFT</td>
<td>Tag1 Tag2 WDAND2TAGAFT Word Tag3</td>
<td>Change Tag1 to Tag2 if the word is Word and the second tag after is Tag3.</td>
</tr>
<tr>
<td>WPREVVTAG</td>
<td>Tag1 Tag2 WPREVVTAG Tag3 Word</td>
<td>Change Tag1 to Tag2 if the word is Word and the previous tag is Tag3.</td>
</tr>
<tr>
<td>WDNEXTTAG</td>
<td>Tag1 Tag2 WDNEXTTAG Word Tag3</td>
<td>Change Tag1 to Tag2 if the word is Word and the next tag is Tag3.</td>
</tr>
</tbody>
</table>

Moreover, STAART marks the beginning of the sentence. Thus, the rule

```
  . FN NEXT1OR2TAG STAART
```

means that ‘change the tag . to FN if one of the two following ‘tags’ is the beginning of the sentence.
Contextual rules for PoS tags

STAART . RBIGRAM . STAART
FN STAART LBIGRAM STAART STAART
KOT , PREVTAG ,
FN DET WDPREVTAG DET a
, FN NEXTWD ,
. FN NEXT1OR2TAG STAART
IGE STAART LBIGRAM STAART STAART
HA STAART LBIGRAM STAART STAART
DET STAART PREVTAG STAART
FN MN PREVTAG MN
, IGE NEXTWD ,
MN STAART LBIGRAM STAART STAART
MN FN PREVTAG MN
STAART FN NEXT2WD STAART
FN IGE PREVTAG IGE
NM , CURWD ,
DET IGE PREV1OR2OR3TAG IGE
IGE FN NEXTTAG IGE
KOT STAART PREVTAG STAART
STAART DET NEXT2WD STAART
NM STAART LBIGRAM STAART STAART
HA , CURWD ,
STAART , RBIGRAM , STAART
. IGE NEXT1OR2TAG STAART
IGE HA NEXTTAG IGE
FN DET CURWD az
DET NM NEXTWD DET
DET KOT NEXT1OR2OR3TAG DET
STAART KOT NEXT2WD STAART
FN STAART LBIGRAM STAART STAART
- STAART PREVTAG STAART
MN DET CURWD a
FN , CURWD ,
, NM NEXTWD ,
IK IGE PREV1OR2OR3TAG IGE
IGE NM NEXTTAG IGE
NM IGE PREVTAG IGE
FN DET CURWD A
DET , CURWD ,
IGE KOT NEXTTAG IGE
DET HA NEXT1OR2OR3TAG DET
HA KOT PREV1OR2OR3TAG ,
HA FN PREV1OR2OR3TAG STAART
IGE - CURWD -
MN , CURWD ,
NM KOT PREV1OR2OR3TAG ,
KOT FN PREV1OR2TAG FN
IGE FN NEXTTAG IGE
, MN NEXTWD ,
STAART ? RBIGRAM ? STAART
DET FN NEXTTAG DET
NU FN PREV1OR2OR3TAG STAART
KOT FN PREV1OR2OR3TAG DET
MN DET CURWD az
  . NM NEXTTAG .
  , HA NEXTTAG ,
HA KOT PREV1OR2OR3TAG STAART
HA IGE PREVTAG IGE
IGE HA NEXTTAG IGE
MN FN NEXTTAG MN
FN NM SURROUNDTAG NM FN
NM FN PREV1OR2OR3TAG STAART
FN INF PREVTAG INF
IGE MN SURROUNDTAG MN IGE
MN HA NEXTTAG MN
MN DET NEXTTAG MN
FN SZN SURROUNDTAG SZN FN
STAART ! RBIGRAM ! STAART
IGE , CURWD ,
STAART MN NEXT2WD STAART
STAART HA NEXT2WD STAART
STAART NM NEXT2WD STAART
STAART IGE NEXT2WD STAART
MN STAART LBIGRAM STAART STAART
MN KOT NEXTTAG MN
HA NM NEXTTAG HA
NM HA NEXTTAG NM
HA KOT CURWD is
DET KOT PREV1OR2OR3TAG ,
FN DET CURWD egy
SZN STAART LBIGRAM STAART STAART
HA FN PREV1OR2OR3TAG DET
- FN NEXTWD -
FN KOT CURWD hogy
INF STAART LBIGRAM STAART STAART
HA - CURWD -
  , INF NEXTWD ,
FN KOT CURWD eis
INF IGE NEXTTAG INF
IGE FN NEXTTAG IGE
FN KOT CURWD ha
  . NU NEXTTAG .
  . MN NEXTTAG .
MN NM NEXTTAG MN
NM DET NEXTTAG MN
FN - CURWD -
NU FN PREV1OR2OR3TAG STAART
FN MN SURROUNDTAG MN FN
IK IGE PREV1OR2OR3TAG STAART
KOT MN PREV1OR2TAG MN
FN DET CURWD Az
  . INF NEXTWD .
FN IGE NEXTWD volna
MN IGE PREVTAG IGE
MN FN NEXTTAG MN
MN HA NEXTTAG MN
  . IK NEXTTAG .
IGE HA CURWD nem
NM - CURWD -
MN FN PREVBIGRAM MN FN
DET NM NEXTTAG DET
KOT IK CURWD meg
, IK NEXTWD ,
! FN NEXTWD !
KOT - CURWD -
? IGE NEXTWD ?
KOT NM NEXTTAG KOT
DET HA NEXTTAG DET
DET IGE PREV1OR2TAG STAART
KOT HA NEXTWD is
IGE STAART LBIGRAM STAART STAART
, NU NEXTTAG ,
NM FN PREV1OR2TAG FN
IGE FN PREVTAG DET
FN IK CURWD fel
: FN NEXTWD :
FN STAART LBIGRAM STAART STAART
FN NU CURWD ellen
DET MN NEXTTAG DET
MN KOT CURWD els
DET - CURWD -
. HA NEXTTAG .
FN HA CURWD mikor
NM IK CURWD ki
MN DET NEXTTAG MN
? FN NEXTWD ?
HA FN PREVTAG FN
HA NU PREV1OR2OR3TAG FN
HA KOT PREV1OR2OR3TAG FN
SZN DET CURWD a
DET FN NEXTTAG DET
NM KOT NEXTTAG NM
DET KOT PREV1OR2OR3TAG STAART
; IGE NEXTWD ;
NM HA NEXT1OR2OR3TAG IGE
- IGE NEXTWD -
- ! NEXTWD -
, - NEXTWD ,
DET NM NEXTTAG IGE
HA IK PREVTAG IGE
KOT FN NEXTTAG KOT
INF IGE NEXTTAG INF
HA ; CURWD ;
DET IK PREV1OR2OR3TAG IGE
FN IGE SURROUNDTAG STAART NM
FN NM SURROUNDTAG , FN
FN KOT SURROUNDTAG , FN
NM FN PREV1OR2OR3TAG DET
IGE IK PREVTAG IK
NM IGE PREV1OR2TAG STAART
IGE INF PREVTAG INF
KOT IGE PREVTAG IGE
IGE MN SURROUNDTAG MN IGE
IGE HA NEXTBIGRAM IGE STAART
FN KOT CURWD is
MN KOT CURWD hogy
STAART - RBIGRAM - STAART
HA DET NEXT1OR2OR3TAG FN
NM INF PREVTAG INF
" FN NEXTWD "
FN NM CURWD eln
MN NM NEXTTAG MN
NM MN CURWD valamilyen
MN SZN NEXTTAG MN
MN FN PREVTAG MN
INF FN NEXTTAG INF
INF HA NEXTTAG INF
DET : CURWD :
FN IGE PREVTAG IGE
IGE FN PREVBIGRAM IGE IGE
IGE FN PREVBIGRAM NM IGE
IGE FN PREVBIGRAM DET MN
FN HA SURROUNDTAG HA FN
HA NM NEXTTAG HA
IGE FN SURROUNDTAG STAART IGE
IGE HA SURROUNDTAG STAART IGE
IGE MN SURROUNDTAG STAART IGE
FN SZN SURROUNDTAG SZN FN
NM DET NEXTTAG FN
DET MN NEXT1OR2OR3TAG FN
NM HA NEXTTAG NM
HA MN NEYTTAG HA
SZN FN PREV1OR2OR3TAG DET
INF IGE NEXTTAG INF
SZN KOT PREVTAG ,
INF MN NEXTTAG INF
NU NM PREV1OR2OR3TAG STAART
MN - CURWD -
KOT HA CURWD mielrt
DET NM NEXTWD a
USZ DET NEXT1OR2OR3TAG FN
STAART " RBIGRAM " STAART
FN " WDNEXTTAG " FN
? NM NEXITWD ?
, SZN NEXTWD ,
SZN HA NEXT1OR2TAG SZN
MN FN NEXTTAG MN
IGE KOT WPREVTAG IGE vagy
IK HA NEXTTAG IK
HA KOT CURWD is
HA IGE PREV1OR2OR3TAG STAART
KOT MN NEXT1OR2TAG KOT
FN DET LBIGRAM STAART Egy
DET INF NEXTTAG DET
- ? NEXTTAG -
! IGE NEXTTAG !
IGE MN PREVWD a
IGE NM NEXTBIGRAM IGE STAART
IGE MN SURROUNDTAG MN IGE
DET NM NEXTTAG ,
IGE NM NEXTBIGRAM IGE ?
FN KOT WDNEXTTAG vagy FN
IK KOT NEXT1OR2TAG IK
KOT IK PREV1OR2OR3TAG STAART
IK NM PREVTAG STAART
NM DET NEXTBIGRAM IGE FN
NU FN NEXTTAG NU
FN HA SURROUNDTAG , FN
HA KOT CURWD sem
IGE DET CURWD a
DET HA NEXTTAG DET
SZN NM NEXTTAG SZN
HA DET NEXTTAG HA
STAART ... RBIGRAM ... STAART
- HA NEXTTAG -
HA IK PREV1OR2OR3TAG IK
IK HA NEXT1OR2TAG IK
! SZN NEXTTAG!
FN MN SURROUNDTAG MN FN
FN IGE NEXTWD volna
IGE FN PREVWD volt
IGE FN NEXTBIGRAM IGE KOT
DET NM NEXTWD az
IGE KOT SURROUNDTAG , IGE
IGE KOT CURWD sem
KOT NM NEXTTAG KOT
IGE NU SURROUNDTAG FN IGE
NU IGE PREV1OR2OR3TAG FN
DET FN PREV1OR2TAG STAART
; FN NEXTWD;
. SZN NEXTWD.
" : NEXTWD "
IGE FN SURROUNDTAG MN IGE
FN NM CURWD minden
KOT HA PREV1OR2OR3TAG.
MN HA CURWD nem
IK NM PREVWD,
IGE SZN SURROUNDTAG SZN IGE
SZN KOT NEXT1OR2TAG SZN
KOT FN NEXTTAG KOT
SZN DET NEXT1OR2TAG SZN
KOT NU PREV1OR2OR3TAG STAART
SZN , CURWD,
NM : CURWD : KOT; PREV1OR2OR3TAG FN
FN STAART LBIGRAM STAART STAART
: NM NEXTWD:
NM DET NEXTTAG NM
" MN NEXTWD "
FN MN PREVBIGRAM MN KOT
IGE MN NEXTBIGRAM FN STAART
FN HA PREVBIGRAM FN IGE
FN NM NEXTBIGRAM FN NM
FN NM CURWD olyan
IGE HA NEXT2WD is
KOT HA NEXTWD vagy
MN NM CURWD valami
IGE KOT WDNEXTTAG is IGE
MN KOT CURWD is
FN USZ WDNEXTTAG B FN
NM FN NEXT1OR2OR3TAG NM
HA NU PREVTAG NU
IK FN NEXTTAG IK
IK NU NEXT1OR2OR3TAG STAART
NM NU NEXTTAG NM
KOT SZN NEXTTAG KOT
SZN IGE NEXTTAG SZN
STAART ; RBIGRAM ; STAART
MN : CURWD :
INF , CURWD ,
HA " CURWD "
DET NU NEXTTAG DET
DET KOT NEXTTAG DET
; HA PREV1OR2OR3TAG FN
. ROV PREVTAG ROV
, " NEXTWD ,
" . NEXTTAG "
FN MN PREVBIGRAM MN ,
FN HA WDNEXTTAG nem FN
FN HA WDNEXTTAG melig FN
FN HA CURWD mindig
IGE FN SURROUNDtag IGE INF
FN HA CURWD halt
FN NM CURWD Minden
IGE HA NEXTBIGRAM IGE INF
KOT HA WDNEXTTAG amikor IGE
HA NM NEXTTAG HA
NM HA PREVTAG ,
DET NM NEXTTAG KOT
HA FN PREV1OR2TAG DET
IGE NU SURROUNDtag NU IGE
SZN FN PREVTAG SZN
NU FN NEXTTAG NU
SZN FN NEXTTAG SZN
NU NM NEXTTAG NU
SZN - CURWD -
STAART : RBIGRAM : STAART
MN ; CURWD ;
IK , CURWD ,
IGE ; CURWD ;
HA INF PREVTAG INF
INF IGE NEXTTAG INF
HA : CURWD :
FN ; WDNEXTTAG ; FN
DET ; NEXTTAG DET
- NM NEXTTAG -
- MN NEXTTAG -
, KOT NEXTTAG ,
" KOT NEXTWD "
FN MN NEXTWD va1lt
FN IGE PREVWD nem
FN IGE SURROUNDtag HA DET
FN IGE SURROUNDtag KOT DET
IGE FN NEXT1OR2WD volt
IGE FN PREV1OR2WD nincs
FN HA CURWD csak
FN MN SURROUNDTAG HA FN
FN HA CURWD reiszben
FN NM WDNEXTTAG nelhalny FN
FN NM CURWD valami
MN IGE CURWD iírt
MN HA NEXTBIGRAM HA IGE
IGE HA CURWD Nem
KOT HA NEXTWD azelrt
IGE NM SURROUNDTAG , IGE
DET NM NEXXTAG .
MN NM CURWD olyan
NM HA NEXXTAG NM
IGE KOT CURWD els
IK KOT PREVWD uljra
NM KOT NEXT2TAG FN
KOT FN PREVTAG MN
MN DET LBIGRAM STAART A
IK IGE NEXT1OR2OR3TAG KOT
IGE HA NEXTBIGRAM IGE KOT
INF NM NEXXTAG INF
DET HA NEXTWD egy
INF KOT NEXXTAG INF
MN " CURWD "
KOT " NEXXTAG KOT
DET " NEXXTAG DET
? INF NEXXTAG ?
: IGE NEXTWD :
. " NEXXTAG .
" DET NEXTWD "
" , NEXXTAG "
! NM NEXXTAG !
FN MN PREVWD teljesen
FN MN CURWD gyengelden
FN IGE CURWD faljt
IGE FN PREVWD kelszu2lt
IGE FN PREVWD Az
IGE FN PREV1OR2WD van
IGE FN PREVBIGRAM MN MN
FN HA WDNEXTTAG to2bbel FN
FN HA CURWD este
MN FN WDPREVTAG DET rigol
MN FN PREVBIGRAM MN MN
FN NM WDNEXTTAG egyik FN
FN NM WDNEXTTAG ilyen FN
MN IGE WDPREVTAG FN adott
MN IGE NEXT2TAG INF
FN SZN NEXTWD elv
MN HA LBIGRAM STAART Ro2gto2n
MN HA NEXTBIGRAM MN IGE
MN HA NEXTBIGRAM MN DET
KOT HA NEXTWD azonban
IGE HA NEXTBGRAM IGE MN
HA NM WDAND2TAGAFT benne FN
IGE NM NEXTWD volt
FN KOTWDNEXTTAG Ha FN
HA FN SURROUNDTAG HA FN
IK HA NEXT1OR2WD hogy
FN IK WDNEXTTAG meg FN
NM HA WDNEXTTAG Mielr t HA
HA IK CURWD le
MN KOT CURWD s
IK KOT PREV1OR2TAG DET
FN IK CURWD Fel
MN SZN CURWD számtalan
KOT NM NEXTBIGRAM FN FN
NM IK WDNEXTTAG el NM
IGE IK LBIGRAM STAART Meg
NM FN NEXTTAG MN
FN NM CURWD Magaltoll
IGE IK CURWD el
INF IGE PREVTAG IK
NM KOT NEXTTAG NM
DET SZN NEXT1OR2WD ketto3
IGE DET LBIGRAM STAART Az
NM IGE PREVTAG HA
MN DET LBIGRAM STAART Az
NU IGE NEXT1OR2OR3TAG STAART
ROV STAART PREVTAG STAART
NM " NEXTTAG NM
IGE : WDNEXTTAG : IGE
IGE " CURWD "
FN : WDNEXTTAG : FN
FN . WDNEXTTAG . FN
: KOT NEXTTAG :
- : NEXTWD -
" NEXTTAG -
" IGE NEXTTAG "
! HA NEXTTAG !
HA KOT PREVTAG FN
KOT FN PREV2WD hogy
FN MN NEXTWD laltszo1
FN MN NEXTWD lesz
FN MN NEXTWD laltszott
FN MN NEXTWD voltak
MN FN PREVBIGRAM HA MN
FN MN PREVWD nagyon
FN MN WDNEXTTAG kelszu2lo3ben IGE
FN MN WDPREVTAG FN folyol
FN MN WDPREVTAG MN hatalmasabb
FN MN WDPREVTAG MN jo2vo3
MN FN SURROUNDTAG DET MN
FN MN NEXT1OR2WD csopoort
FN MN CURWD kemeln yen
FN IGE PREVTAG NU
IGE MN NEXTBIGRAM FN IGE
FN IGE NEXTWD ral
MN KOT NEXTBIGRAM MN IGE
FN IGE NEXT1OR2WD mit
FN IGE PREVBIGRAM STAART -
IGE HA NEXTBIGRAM IGE HA
IGE FN PREVBIGRAM , IGE
IGE MN SURROUND TAG , IGE
IGE KOT NEXTBIGRAM IGE HA
FN IGE CURWD hozzaleírt
FN IGE CURWD tudta
FN IGE CURWD tesz
FN IGE CURWD gondolok
IGE FN PREVWD to2rteint
IGE FN PREVWD olyan
IGE FN PREVBIGRAM NU IGE
IGE FN CURWD O'Brien
FN HA LBIGRAM STAART Esetleg
FN HA LBIGRAM STAART Melg
FN HA WDNEXTTAG ma IGE
FN HA WDNEXTTAG sebtiben IGE
FN HA WDNEXTTAG csattanva IGE
FN HA PREVBIGRAM IK IGE
FN HA CURWD malsfelo3l
FN HA CURWD valahonnan
MN FN PREVWD uij
MN FN RBIGRAM Disznol !
MN FN WDNEXTTAG o2nkelintes FN
MN FN SURROUND TAG MN IGE
MN FN WDAND2BFR STAART uljbeszél
IGE MN PREVTAG DET
MN FN PREV1OR2WD mint
IGE MN PREVWD voltak
FN NM LBIGRAM STAART Ebbo3l
FN NM LBIGRAM STAART Ne1ha1ny
FN NM WDNEXTTAG maisik FN
FN NM WDNEXTTAG e FN
FN NM WDNEXTTAG milyen FN
FN NM WDPREVTAG HA kinek
FN NM CURWD mais
FN NM CURWD mindenfelle
FN NM CURWD o2nmagad
FN NM CURWD magukhoz
MN IGE NEXTWD elo3
MN IGE NEXTWD egyetlen
MN IGE LBIGRAM STAART kellett
MN IGE WDPREVTAG FN vetett
MN HA SURROUND TAG DET MN
MN HA CURWD szilvesen
IGE HA PREVWD vannak
KOT HA NEXTWD ha
KOT HA WDNEXTTAG amint HA
KOT HA WDNEXTTAG amikor HA
KOT HA PREVBIGRAM , KOT
IGE NM WDNEXTTAG azt IGE
HA NM CURWD minden
IGE NM WDNEXTTAG Mit IGE
KOT HA NEXTBIGRAM KOT IGE
HA KOT CURWD mint
IGE NM NEXTBIGRAM IGE MN
DET NM NEXTWD bizonyos
FN KOT WDNEXTTAG sem FN
FN KOT CURWD ha
FN KOT CURWD s
HA MN PREVWD egy
HA MN NEXTIOR2WD benne
FN NU PREVTAG NU
HA KOT WAND2TAGAFT amikor HA
HA FN SURROUNDTAG MN HA
FN USZ NEXTWD neveztelek
FN NU CURWD alattuk
KOT IGE PREVWD nem
IGE KOT CURWD mert
NM DET NEXTBIGRAM HA MN
HA IK WDNEXTTAG ra1 IGE
STAART NU RBIGRAM elo3tt STAART
INF FN PREVTAG INF
IK HA CURWD egyu2tt
IK KOT PREVTAG SZN
FN USZ LBIGRAM a duplagondol
HA IK WDPREVTAG KOT egyu2tt
FN IK WDPREVTAG IGE fel
NM MN NEXTBIGRAM NM
MN KOT CURWD sem
FN IK CURWD velgig
SZN HA PREVTAG ,
HA SZN CURWD sokkal
KOT FN PREVTAG MN
NM MN WDNEXTTAG mai1s IGE
HA IGE NEXTW volna
STAART IK RBIGRAM el STAART
STAART IGE PREVBIGRAM STAART STAART
IGE STAART LBIGRAM STAART STAART
IGE FN SURROUNDTAG STAART KOT
STAART SZN NEXTBIGRAM STAART STAART
STAART NU NEXTBIGRAM STAART STAART
STAART INF NEXTBIGRAM STAART STAART
INF IGE NEXTBIGRAM INF
FN IK LBIGRAM STAERT El
SZN HA NEXTW SZN
INF MN NEXTW INF
SZN NM NEXTW SZN
NM DET NEXTW SZN
NU MN NEXTW NU
KOT IK PREVTAG IGE
IK SZN NEXTW meg
IGE DET WDNEXTTAG egy IGE
DET HA NEXTW NM
STAART IK NEXTBIGRAM STAART STAART
MSZ - PREV1OR2TAG STAART
MN INF PREVTAG INF
KOT INF NEXTW KOT
? MN NEXTW ?
IGE KOT NEXTBIGRAM IGE MN
? HA NEXTW ?
HA IK WDPREVTAG IGE ide
; MN NEXTW ;
lesznek
Contextual rules for PoS and subtags

DET NM NEXTTAG DET
DET NM NEXTTAG ,
DET NM NEXTWD volt
DET NM NEXTTAG KOT
MN FN SURROUNDTAG DET ,
IGE_Me3 MN NEXTTAG FN_PL
MN_DAT FN_DAT PREV1OR2OR3TAG DET
MN FN WDPREVTAG DET u1jbeszelle
HA KOT LBIGRAM , amikor
IGE_t3 FN_DAT PREV1OR2TAG DET
IK KOT PREVWD u1jra
MN IGE_Te3 NEXTTAG DET
HA IK PREVBIGRAM HA IGE_Me3
FN_ELA FN_PSe3_ELA PREV1OR2OR3TAG FN
NM_ALL IK PREVTAG IGE_TMe3
KOT HA CURWD mélgiscsak
DET NM NEXTTAG NU
FN_DAT MN_DAT NEXTWD laltszol
FN_INE FN_PSe3_INE PREV1OR2TAG FN
FN_ABL FN_PSe3_ABL PREV1OR2TAG FN
IGE_t1 FN_PSt1 PREV1OR2TAG MN
FN_FOR HA PREV1OR2TAG DET
IGE_Me3 FN_CAU PREVTAG DET
FN_ILL FN_PSe1_ILL CURWD eszembe
IGE_Me3 MN PREVTAG DET
IGE_Me3 MN NEXTTAG FN_PL_INS
IGE_Me3 MN NEXT1OR2WD volt
IGE_Me3 MN NEXTBIGRAM MN FN
MN FN SURROUNDTAG MN IGE_Me3
MN FN SURROUNDTAG DET .
DET NM NEXTTAG .
NU HA CURWD belo3le
IGE_Te3 FN_PSe3 NEXT1OR2OR3TAG IGE_Me3
NNP FN NEXT1OR2OR3TAG .
IGE FN PREVTAG DET
IGE_INRt3 FN_PSt3 PREVTAG MN
IGE_Pe3 FN_SUP PREVTAG MN
IGE_t1 IGE_INRt1 NEXT1OR2TAG IGE
MN FN NEXTTAG NU
FN_DAT IGE_t3 PREV1OR2OR3TAG FN_PL
HA FN PREVWD a
FN_PSe3_INS FN_INS PREVTAG DET
FN_ELA FN_PL_ELA SURROUNDTAG , ,
FN HA CURWD dellutaln
MN_DAT HA NEXT1OR2TAG KOT
IGE_Tt1 FN_PSt3 PREVTAG DET
IGE_TMe2 IGE PREV1OR2OR3TAG FN
IGE_TMt3 FN CURWD bizonyíitelk
NNP FN NEXT1OR2OR3TAG FN
FN_DEL HA CURWD haltulroll
FN_ADE FN_PSe3_ADE PREVTAG FN
IGE_TPt3 FN PREV1OR2TAG DET
FN MN CURWD furcsa
KOT HA NEXTWD csak
KOT HA WDNEXTTAG amikor NM
KOT HA CURWD hacsak
KOT HA CURWD ugyancsak
FN_DAT IGE_t3 PREVTAG FN_ACC
FN_ACC FN NEXTTAG NU
MN_ESS_MOD HA CURWD e1ppolyan
FN_PSe3 FN PREV1OR2WD egy
FN_DAT MN_DAT PREV1OR2TAG IGE_TMe3
IGE_Tmt3 FN_PL PREVTAG MN
IK KOT SURROUNDTAG HA DET
FN_DAT MN_DAT NEXTBIGRAM IGE_Me3,
IGE_Tmt3 FN_PL PREVTAG IGE_Me3
MN_PL FN_PL PREVTAG MN
FN_ABL FN_PSe3_ABL NEXT1OR2TAG KOT
IGE_t3 FN_DAT PREVTAG MN
FN_ILL FN_PSe3_ILL CURWD hatalmalba
IGE_t1 FN_PSt1 NEXT1OR2OR3TAG NU
IGE_Me3 FN_ACC NEXTTAG IGE_Me3
FN_SUP FN CURWD szexbu3n
FN_PSe3 MN CURWD tompa
FN_ABL FN_PL_ABL PREV1OR2OR3TAG ,
IGE_TMe3 NU CURWD fo21o2tte
FN_PSt3_ACC FN_PL_ACC PREV1OR2OR3TAG DET
IGE_TMe3 FN PREVTAG DET
IGE_TMe3_KSZ IGE_Te3_KSZ PREVTAG HA
MN_FOK SZN LBIGRAM STAART To2bb
IGE_MN IGE_Me3 CURWD igyekezett
FN_INS IK CURWD Fel
IGE_Me3 MN NEXTTAG FN_INE
IGE_Me3 MN NEXTTAG FN_PL_ACC
IGE_Me3 MN NEXT1OR2TAG IGE
IGE_Me3 MN NEXT1OR2OR3TAG FN_PSe3_INS
IGE_Me3 MN NEXTBIGRAM MN MN
MN FN NEXTTAG -
MN FN LBIGRAM a proletarialtus
FN MN NEXTTAG FN_SUP
MN FN NEXTBIGRAM KOT DET
FN MN NEXTBIGRAM , MN
MN FN CURWD o2nkelnintes
FN_PSe3_ACC FN_ACC PREV1OR2TAG NM_ACC
FN_PSe3_ACC FN_ACC NEXT1OR2OR3TAG IGE
FN MN NEXTTAG FN_ACC
FN_PSe3_ACC FN_ACC PREVBIGRAM DET MN
KOT HA LBIGRAM , amilig
FN_PSe3_ACC FN_ACC PREV2TAG IK
DET NM NEXTTAG ?
FN_PL MN_PL CURWD gazdagok
FN_ACC FN WDPREVTAG DET drolt
HA KOT WDNEXTTAG amilig IK
HA FN WDPREVTAG MN konyha
FN_DAT IGE_t3 NEXTTAG NM_ACC
IGE_e3 IGE_Tt3 NEXT1OR2TAG NM_ACC
HA KOT CURWD illeto3leg
HA IK WDNEXTTAG ral IGE
IGE_e3 IGE_Tt3 PREV1OR2TAG FN_PL
HA NM_INE RBIGRAM benne,
Appendix D: Text normalisation

Normalisation of the training corpus (Orwell’s 1984)

NORMSUT.BAT
@ECHO OFF

pcbeta -#to#.rul orwdis1.txt -#to#ut.txt
pcbeta #bort.rul -#to#ut.txt #bortut.txt
pcbeta stj0{.rul #bortut.txt stj0{ut.txt
pcbeta tomrad.rul stj0{ut.txt trut.txt
pcbeta delim.rul trut.txt delimit.txt
pcbeta fn.rul delimit.txt fnut.txt
pcbeta usz.rul fnut.txt uszut.txt
pcbeta mn.rul uszut.txt mnut.txt
pcbeta ige.rul mnut.txt igeut.txt
pcbeta chap.rul igeut.txt chaput.txt
pcbeta oms0.rul chaput.txt oms0ut.txt
pcbeta oms1.rul oms0ut.txt oms1ut.txt
pcbeta clip1.rul oms1ut.txt cli1ut.txt
pcbeta oms2.rul cli1ut.txt oms2ut.txt
pcbeta clip2.rul oms2ut.txt clip2ut.txt
pcbeta oms3.rul clip2ut.txt oms3ut.txt
pcbeta clip3.rul oms3ut.txt clip3ut.txt
pcbeta oms4.rul clip3ut.txt oms4ut.txt
pcbeta clip4.rul oms4ut.txt clip4ut.txt
pcbeta oms5.rul clip4ut.txt oms5ut.txt
pcbeta oms6.rul oms5ut.txt oms6ut.txt
pcbeta oms7.rul oms6ut.txt oms7ut.txt
pcbeta oms8.rul oms7ut.txt oms8ut.txt
pcbeta oms9.rul oms8ut.txt oms9ut.txt
pcbeta oms10.rul oms9ut.txt oms10ut.txt
pcbeta oms11.rul oms10ut.txt oms11ut.txt
pcbeta oms12.rul oms11ut.txt oms12ut.txt
pcbeta oms13.rul oms12ut.txt oms13ut.txt
pcbeta oms14.rul oms13ut.txt oms14ut.txt
pcbeta clip14.rul oms14ut.txt cli14ut.txt
pcbeta clip15.rul cli14ut.txt cli15ut.txt
pcbeta oms15.rul cli15ut.txt oms15ut.txt
pcbeta clip4.rul oms15ut.txt cli14ut.txt
pcbeta oms15.rul cli14ut.txt oms15ut.txt
pcbeta clip4.rul oms151u.txt cli142ut.txt
pcbeta clip1.rul cli142ut.txt cli1ut.txt
pcbeta oms16.rul cli1ut.txt oms16ut.txt
pcbeta oms18.rul oms16ut.txt oms18ut.txt
pcbeta oms18.rul oms18ut.txt oms181.txt
pcbeta oms18.rul oms181.txt oms182.txt
pcbeta oms18.rul oms182.txt oms183.txt
pcbeta clip18.rul oms183.txt cli18ut.txt
pcbeta menrad.rul cli18ut.txt menraut.txt
pcbeta oms17.rul menraut.txt subtag.txt
del -#to#ut.txt
del #bortut.txt
del stj@ut.txt
del trut.txt
del delimut.txt
del fnut.txt
del uszut.txt
del mnut.txt
del igemut.txt
del chaput.txt
del oms0ut.txt
del oms1ut.txt
del clip1ut.txt
del oms2ut.txt
del clip2ut.txt
del oms3ut.txt
del clip3ut.txt
del oms4ut.txt
del clip4ut.txt
del oms5ut.txt
del oms6ut.txt
del oms7ut.txt
del oms8ut.txt
del oms9ut.txt
del oms10ut.txt
del oms11ut.txt
del oms12ut.txt
del oms13ut.txt
del oms14ut.txt
del oms15ut.txt
del oms16ut.txt
del oms17ut.txt
del oms18ut.txt
del oms181.txt
del oms182.txt
del oms183.txt
del cli18ut.txt
del menraut.txt

-#to#.rul
  _-#Numeral.Numeral => #Numeral.Numeral in notes

CHARSET
Dig: 48-57

RULES         LC    RC    SC    RS
-#; #;        32    Dig    1     2

#bort.rul
  _cut #Numeral.Numeral in notes

STATESET
Cut: 10
CHARSET
Dig: 48-57

RULES | LC | RC | SC | RS | MV
---|---|---|---|---|---
#;; | 32 | -# | 1 | 2 | -3
#;; | Dig | # | 2 | 10 | 5

stj{.rul
_*Ltr --> Ltr*
_@Ltr --> Ltr*
_delete { between # and letter
_delete } between space and letter
_delete | between space and letter

CHARSET
Ltr: 65-90 97-122 å ä ö Å Ä Ö

RULES | LC | RC | SC | RS
---|---|---|---|---
#*; | #; | Ltr | 1 | 2
#@; | #; | Ltr | 1 | 2
#{ ; | #; | Ltr | 1 | 2
}; | ; | 32 | # | 1 | 2
|; | ; | 32 | # | 1 | 2

tomrad.rul
_delete every empty line

RULES | LC | RC | SC | RS | MV
---|---|---|---|---|---
#;; | # | # | 1 | 2 | 0

delim.rul
stag ./.@ ?/?@, and !/@ for
_marking sentence per entry with @
tag ,/, =/- ;; ((/ )/) and :/:, etc.

CHARSET
Del: /
Ltrrdel: 48-57 65-90 97-122 /

RULES | LC | RC | SC | RS
---|---|---|---|---
.; | /.; | # | # | 1 | 2
*; | */*; | # | # | 1 | 2
;; | ::; | # | # | 1 | 2
(; | /(; | # | # | 1 | 2
-; | -=; | # | # | 1 | 2
+; | +/+; | # | # | 1 | 2
=; | /==; | # | # | 1 | 2
...; | ../...@; | # | # | 1 | 2
.; | ./@; | # | # | 1 | 2
?; | ?/@; | # | -Del | 1 | 2
!; | !/@; | # | -Del | 1 | 2
); | )/@; | # | -Del | 1 | 2
"; | "/@; | # | -Ltrrdel | 1 | 2
fn.rul
_tag the following words as noun

<table>
<thead>
<tr>
<th>RULES</th>
<th>LC</th>
<th>RC</th>
<th>SC</th>
<th>RS</th>
</tr>
</thead>
<tbody>
<tr>
<td>%Aaronson; aaronson[FN]=Aaronson;</td>
<td>#</td>
<td></td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>%Aaronson; aaronson[FN]+t[ACC];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Aaronsonnak; aaronson[FN]+nak[DAT];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Amershambe; Amersham[FN]+be[ILL];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Ampleforth; ampleforth[FN]=Ampleforth;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Ampleforthra; ampleforth[FN]+ra[SUB];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Amplefortht; ampleforth[FN]+t[ACC];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Ampleforthot; ampleforth[FN]+ot[ACC];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Angszoc; angszoc[FN]=Angszoc;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%ANGSZOC; angszoc[FN]=ANGSZOC;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Astaro1th; astaro1th[FN]=Astaro1th;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Astaro1thnak; astaro1th[FN]+nak[DAT];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Baailnak; baail[FN]+nak[DAT];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Bailey; bailey[FN]=Bailey;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Berkhamsted; berkhamsted[FN]=Berkhamsted;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Berkhamstedbe; berkhamsted[FN]+be[ILL];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Brazzaville; brazzaville[FN]=Brazzaville;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Bumstead; bumstead[FN]=Bumstead;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Charrington; charrington[FN]=Charrington;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Charringtont; charrington[FN]+t[ACC];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Charringtonna; charrington[FN]+na[INS];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Charringtonto; charrington[FN]+to[ABL];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Chauser; chauser[FN]=Chauser;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Clement; clement[FN]=Clement;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Colchester; colchester[FN]=Colchester;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Colchesterre; colchester[FN]+re[SUB];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Cromwell; cromwell[FN]=Cromwell;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Danes; danes[FN]=Danes;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Emmanuel; emmanuel[FN]=Emmanuel;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%EMMANUEL; emmanuel[FN]=EMMANUEL;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Gestapo; gestapo[FN]=Gestapo;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Goldstein; goldstein[FN]=Goldstein;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%GOLDSTEIN; goldstein[FN]=GOLDSTEIN;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Goldstein; Goldstein[FN]+t[ACC];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Goldsteinnek; Goldstein[FN]+nek[DAT];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Goldsteinkel; Goldstein[FN]+kel[INS];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Goldsteinso; Goldstein[FN]+so[ABL];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Goldsteinro; Goldstein[FN]+ro[DEL];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Goldsteinre; Goldstein[FN]+re[SUB];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Hyde; hyde[FN]=Hyde;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Inprekor; inprekor[FN]=Inprekor;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Internacional; internacional[FN]=Internacional;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Jefferson; jefferson[FN]=Jefferson;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Julia; julia[FN]=Julia;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Julialt; julia[FN]=Julial+t[ACC];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Juliale1; julia[FN]=Julial+e1[POS];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Julialra; julia[FN]=Julial+ra[SUB];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Julialro; julia[FN]=Julial+ro[DEL];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Julialval; julia[FN]=Julial+val[INS];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Julialnak; julia[FN]=Julial+nak[DAT];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Rules

<table>
<thead>
<tr>
<th>%arcbu3n; arcbu3n[USZ]</th>
<th>#</th>
<th>#</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>%arcbu3n-nek; arcbu3n[USZ]+-nek[DAT]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Avenue-n; Avenue[USZ]+-n[SUP]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%B; b[USZ]=B</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
%times; times[USZ];
%Times-ba; Times[USZ]+-ba[ILL];
%Times-ban; Times[USZ]+-ban[INE];
%Times-boll; Times[USZ]+-boll[ELA];
%to1k; to1k[USZ];
%U1ESZ; U1ESZ[USZ];
%U1ESZ-t; U1ESZ[USZ]+-t[ACC];
%utalnikt; utalnikt[USZ];

**mn.rul**

_tag for fantasy words with normal inflection

<table>
<thead>
<tr>
<th>RULES</th>
<th>LC</th>
<th>RC</th>
<th>SC</th>
<th>RS</th>
</tr>
</thead>
<tbody>
<tr>
<td>%duplapluszt; duplapluszt[MN];</td>
<td></td>
<td></td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>%duplapluszhideg; duplapluszhideg[MN];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%duplapluszjol; duplapluszjol[MN];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%duplaplusznemjol; duplaplusznemjol[MN];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%egyenlo3; egyenlo3[MN];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%jol; jol[MN];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%jolgondolos; jolgondolos[MN];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%jolgondolosan; jolgondolosan[MN]+an[ESSMOD];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%minibo3; minibo3[MN];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%nemhideg; nemhideg[MN];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%nemjol; nemjol[MN];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%nemso2te1lt; nemso2te1lt[MN];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%nemvilalgos; nemvilalgos[MN];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%olbeszellu21; olbeszellu21[MN]+u21[ESSMOD];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%olbeszellu21; Olbeszellu21[MN]+u21[ESSMOD];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%olgondololik; Olgondololik[MN]+k[PL];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%plusz; plusz[MN];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%pluszhideg; pluszhideg[MN];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%pluszjol; pluszjol[MN];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%plusztelejes; plusztelejes[MN];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%sebesselegesen; sebesselegesen[MN]+en[ESSMOD];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%so2te1lt; so2te1lt[MN];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%St; st[MN]=St;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%Saint; saint[MN]=Saint;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%szabad; szabad[MN];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%to2bbese; to2bbes[MN]+e[PSe3];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%uljbeszell; uljbeszell[MN];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%ULjbeszell; ULjbeszell[MN]=ULjbeszell;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%uljbeszellu21; uljbeszellu21[MN]+u21[ESSMOD];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%ULjbeszellu21; ULjbeszellu21[MN]+u21[ESSMOD];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%vilalgos; vilalgos[MN];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%-os; -os[SKEP];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%-es; -es[SKEP];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%-o2s; -o2s[SKEP];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%-osan; -osan[SKEP]+an[ESSMOD];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%-esen; -esen[SKEP]+en[ESSMOD];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%-o2sen; -o2sen[SKEP]+en[ESSMOD];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%elo3; elo3[IK];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%utaln; utaln[NU];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%nagyon; nagyon[HA];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%nem; nem[HA];</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
_following verbs are incorrect in Orwdia.txt

RULES          LC  RC  SC  RS
%aszongya[IGE]; aszongy[IGE]+a[Te3];       #    #    1    2
%aszongya[IGE]=Aszongya; Aszongy[IGE]+a[Te3];
%aszondom[IGE]=Aszondom; Aszond[IGE]+om[Te1];
%aszondom[IGE]; aszond[IGE]+om[Te1];
%haselrez; haselrez[IGE];
%nemhaselrzik; namhaselrez[IGE]=nemhaselrz+ik[e3];
%nemfolytatni; nemfolytatni[INF];
%jolgondolt; jolgondol[IGE]+t[Me3];
%gondol; gondol[IGE];
%valg; valg[IGE];

chap.rul
_delete the following lines:
_%hi
_%/hi
_%/div
_%div
_%type=chapter
_%type=part
_%n=1/2/3/4/5/6/7/8/9
_%/p
_%p
_> 
_<

RULES          LC  RC  SC  RS  MV
#%hi; ;       #    #    1    2    0
#%/hi; ;
#%div1; ;
#%div2; ;
#%/div1; ;
#%/div2; ;
#%type=chapter; ;
#%type=part; ;
#%n=1; ;
#%n=2; ;
#%n=3; ;
#%n=4; ;
#%n=5; ;
#%n=6; ;
#%n=7; ;
#%n=8; ;
#%n=9; ;
#%/p; ;
#%p; ;
#>; ;
#<; ;
oms0.rul

DEFTYP
1: 32

RULES
| e1n[NM]=+ra[SUB]=ralm; e1n[NM]=+ral[SUB]=+m; | #   # 1 2 5 |
| te[NM]=+ra[SUB]=rald; te[NM]=+ral[SUB]=+d; | #   # 1 2 5 |
| o3[NM]=+ra[SUB]=ral; o3[NM]=+ral[SUB];    | #   # 1 2 5 |
| o3k[NM]=+ra[SUB]=raljuk; o3k[NM]=+ral[SUB]=juk; | #   # 1 2 5 |

oms1.rul

_word[TAG]=Word => word%Word/WORD[TAG]
_word[TAG]="Word => word"%Word/WORD[TAG]
_word[TAG1]=Word+morph[TAG2] => word%Word+morph[TAG1][TAG2]
_morph[TAG1]=Word+[TAG2]= => word%Word+[TAG1][TAG2]=
_word.[TAG]=Word. => word.%Word./WORD.[TAG]

DEFTYP
1: 32

CHARSET
Maj: 65-90 '

STATESET
Cut: 10
Paste: 20

RULES
| [; %[; | Ltr Maj 1 2 -4 |
| ]=; ]; | Ltr Maj 2 10 5 |
| #; ;   | Ltr  # 10 2 5 |
| [;   ]; | Ltrplus Ltr 10 20 5 |

clip1.rul

_ord%Ord[TAG] => Ord[TAG]
_ord%"Ord[TAG] => °Ord[TAG]

DEFTYP
1: 32

CHARSET
Majcit: 65-90 "

STATESET
Cut: 10

RULES
| #; %;   | Ltr Majcit 2 10 5 |
oms2.rul

_morph%[TAG1]=ord => morph%ord[[TAG1]]

DEFTYP
1: 32

CHARSET
Ltr: 48-57 65-90 97-122
Maj: 65-90
Plus: +
Ltrplus: 48-57 65-90 97-122 + 

STATESET
Cut: 10
Paste: 20

RULES  LC  RC  SC  RS  MV
%[; %[; Ltr  Maj  1  2  -4
]=; ]; Ltr  -Plus  2  10  5
[; ; Ltrplus Ltr  10  20  5
#; ; Ltr  #  10  20  5

clip2.rul

_word%word[[TAG1]TAG2] => word[[TAG1]TAG2]
_word%word+morph[[TAG1]TAG2] => word+morph[[TAG1]TAG2]
_word"word+morph"[[TAG1]TAG2] => "word+morph"[[TAG1]TAG2]

DEFTYP
1: 32

CHARSET
Ltr: 48-57 65-90 97-122
Ltrcit: 48-57 65-90 97-122 

STATESET
Cut: 10

RULES  LC  RC  SC  RS  MV
#; #; # Ltr  1  2  -3
%; ; Ltr  Ltrcit  2  10  5

oms3.rul

_word%[TAG1]+morph[TAG2] => wordmorph[[TAG1]TAG2]
STATESET
Cut: 10
Paste: 20

RULES  LC  RC  SC  RS  MV
%[; ]  Ltr  Maj  1  2  -4
]+; ];  Ltr  -Del  2  10  5
[; ];  Ltr  Ltr  10  20  5

clip3.rul
_word+morph[LONGTAG1]TAG2] => word+morph[LONGTAG1]TAG2]
_word+"morph1"[LONGTAG1]TAG2]+-morph2 => word+morph[LONGTAG1]TAG2]

DEFTYP
1: 32

CHARSET
Ltr: 48-57 65-90 97-122
Ltrcit: 48-57 65-90 97-122 *
Ltrplus: 48-57 65-90 97-122 +

STATESET
Cut: 10

RULES  LC  RC  SC  RS  MV
[; ;  Ltrcit  Ltr  1  2  -5
]; ;  Ltr  Ltrplus  2  1  5

oms4.rul
_word[LONGTAG1]+morph[TAG2] => wordmorph[LONGTAG1]TAG2]
_word[LONGTAG1]+-morph[TAG2] => word- morph[LONGTAG1]TAG2]
_word.%[LONGTAG1]+-morph[TAG2] => word.-morph[LONGTAG1]TAG2]

DEFTYP
1: 32

CHARSET
Ltrdel: 48-57 65-90 97-122 -
Maj: 65-90

STATESET
Cut: 10
Paste: 20

RULES  LC  RC  SC  RS  MV
.%[; .[; Ltr  Maj  1  2  -4
]+; ];  Ltr  Ltrdel  2  10  5
[; ];  Ltr  Ltr  10  20  5

clip4.rul
_word+morph[LONGTAG1]TAG2] => word+morph[LONGTAG1]TAG2]
_word+[LONGTAG1]TAG2]= => word[LONGTAG1]TAG2]=

DEFTYP
1: 32
CHARSET
Ltr: 48-57 65-90 97-122
Maj: 65-90

STATESET
Cut: 10

RULES    LC   RC   SC   RS   MV
[][]     Ltr   Ltr  1    2    -5
++[]  Ltr   Ltr  1    2    -5
[];    Ltr   Ltr  2    1    5

oms5.rul
_morph[TAG1]=+[TAG2]=+=word => morph([TAG1]TAG2)=+=word

DEFTYP
1: 32

CHARSET
Ltr: 48-57 65-90 97-122
Maj: 65-90

STATESET
Cut: 10
Paste: 20

RULES    LC   RC   SC   RS   MV
[];     Ltr   Ltr  1    2    -4
[]=+;   Ltr   [   2    10   5
[];     Ltr   Ltr  10   20    5

oms6.rul
_morph1[TAG1]=+morph2[TAG2]=+morph3 => morph=word([TAG1]TAG2)+=morph3

DEFTYP
1: 32

CHARSET
Ltr: 48-57 65-90 97-122
Maj: 65-90

STATESET
Cut: 10
Paste: 20

RULES    LC   RC   SC   RS   MV
[];     Ltr   [   1    2    -4
[]=+;   Ltr   =   2    10    5
#;     Ltr   #   10   20    5

oms7.rul
_morph1[TAG1]=+morph2[TAG2]=+morph3 => morph=word([TAG1]TAG2)+=morph3

DEFTYP
1: 32
CHARSET
Ltr: 48-57 65-90 97-122
Maj: 65-90

STATESET
Cut: 10
Paste: 20

RULES       LC    RC   SC   RS   MV
[; ]       Ltr   Ltr   1    2   -4
[; ]+=; ]=; Ltr   Ltr   2    10   4
[; ]       Ltr   Ltr  10   20   5

oms8.rul
_morph%word[\[TAG1\]TAG2]+=morph3 => morph%word+=morph3[\[TAG1\]TAG2]

DEFTYP
1: 32

CHARSET
Ltr: 48-57 65-90 97-122

STATESET
Cut: 10
Paste: 20

RULES       LC    RC    SC    RS     MV
[; ]       Ltrdel   Maj    1     2     -4
[; ]+=; ]=; Ltr      Del    2    10      5
[; ]       Ltr      Ltr  10    20      5

oms9.rul
_word[USZ]+-morph[\[TAG2\]] => word-morph[\[TAG1\]TAG2]

DEFTYP
1: 32

CHARSET
Ltr: 48-57 65-90 97-122
Ltrdel: 48-57 65-90 97-122
Maj: 65-90
Del: -

STATESET
Cut: 10
Paste: 20

RULES       LC    RC    SC    RS     MV
[; ]       Ltrdel   Maj    1     2     -4
[; ]+=; ]=; Ltr      Del    2    10      5
[; ]       Ltr      Ltr  10    20      5

oms10.rul
_Word+=morph[\[TAG1\]] => Wordmorph[\[TAG1\]]
_Word+=morph=word[\[TAG1\]TAG2] => Wordmorph[\[TAG1\]TAG2]
DEFTYP
1: 32

CHARSET
Ltr: 48-57 65-90 97-122
Maj: 65-90

STATESET
Cut: 10

RULES       LC   RC   SC   RS   MV
#; #;        Maj   1    2    5
+=; ;        Ltr   Ltr   2    3    5
=; ;         Ltr   Ltr   3    4   -5
[; [;        Ltr   [     4    10   4

oms11.rul
_word[TAG1]+=morph => wordmorph[TAG1]

DEFTYP
1: 32

CHARSET
Ltr: 48-57 65-90 97-122

STATESET
Cut: 10
Paste: 20

RULES       LC   RC   SC   RS   MV
[; [;        Ltr   Ltr   1    2   -4
]++; );      Ltr   Ltrdel  2   10    5
#; #        Ltr   #     10   20    5

oms4.rul
_word[LONGTAG1]+morph[TAG2] => wordmorph[[LONGTAG1]TAG2]
_word[LONGTAG1]+-morph[TAG2] => word-morph[[LONGTAG1]TAG2]
_word.%[LONGTAG1]+-morph[TAG2] => word.-morph[[LONGTAG1]TAG2]

DEFTYP
1: 32

CHARSET
Ltrdel: 48-57 65-90 97-122 -
Maj: 65-90

STATESET
Cut: 10
Paste: 20

RULES       LC   RC   SC   RS   MV
.%[; .[;    Ltr   Maj   1    2   -4
]++; );      Ltr   Ltrdel  2   10    5
[; [;        Ltr   Ltr   10   20    5
oms12.rul

DEFTYP
1: 32

RULES | LC | RC | SC | RS | MV
--- | --- | --- | --- | --- | ---
enye1m=et; enye1met; # | 1 | 2 | 5
enyelm=hez; enyelmhez;
o2vel=hez; o2velhez;
o2vel=nek; o2velnek;
o2vel=be; o2velbe;
o2vel=to3l; o2velto3l;
o2vel=t; o2velt;
o2vel=re; o2velre;
mie1nk=hez; mie1nkhez;

oms13.rul

_#word[TAG]=# => #word[TAG]#

DEFTYP
1: 32

CHARSET
Ltr: 48-57 65-90 97-122

RULES | LC | RC | SC | RS | MV
--- | --- | --- | --- | --- | ---
; | Ltr | # | 1 | 2 | 5

oms14.rul

_morph=word[TAG] => word[TAG]

DEFTYP
1: 32

CHARSET
Ltr: 48-57 65-90 97-122

STATESET
Cut: 10

RULES | LC | RC | SC | RS | MV
--- | --- | --- | --- | --- | ---
#; #; # Ltr 1 2 -5
=; ; Ltr Ltr 2 10 5

clip14.rul

_word[[TAG1]TAG2] => word[TAG1TAG2]

DEFTYP
1: 32

CHARSET
Ltr: 48-57 65-90 97-122

STATESET
Cut: 10
RULES     LC   RC   SC   RS    MV
[; ;     Ltr  Ltr  1   2   -5
]; ;     Ltr  Ltr  2   1   5

clip15.rul
_CLIP + between letters

DEFTYP
1: 32

CHARSET
Ltr: 48-57 65-90 97-122
Maj: 65-90

STATESET
Cut: 10

RULES     LC   RC   SC   RS    MV
+; ;     Ltr  Ltr  2   10   5

oms15.rul

DEFTYP
1: 32

CHARSET
Ltr: 48-57 65-90 97-122
Maj: 65-90

STATESET
Cut: 10
Paste: 20

RULES     LC   RC   SC   RS    MV
[; ;     Ltr  Maj  1   2   -4
]+; ];    Ltr  Ltr  2   10   5
[; ;     Ltr  Ltr  10   20   5

clip4.rul
_word+morph[[LONGTAG1]TAG2] => word+ morph[[LONGTAG1]TAG2]
_word+[[LONGTAG1]TAG2]= => word[[LONGTAG1]TAG2]=

DEFTYP
1: 32

CHARSET
Ltr: 48-57 65-90 97-122
Maj: 65-90

STATESET
Cut: 10

RULES     LC   RC   SC   RS    MV
[; ;     Ltr  Ltr  1   2   -5
oms15.rul

_word[LONGTAG1]+morph[TAG2] => wordmorph[[LONGTAG1]TAG2]

DEFTYP
1: 32

CHARSET
Ltr: 48-57 65-90 97-122
Maj: 65-90

STATESET
Cut: 10
Paste: 20

RULES     LC   RC    SC   RS    MV
[; [;     Ltr  Maj    1    2   -4
]; ;    Ltr  Ltr    2   10    5
[; [;     Ltr  Ltr   10   20    5

cipl.rul

_word+[LONGTAG1]TAG2] => word+morph[[LONGTAG1]TAG2]
_word+[LONGTAG1]TAG2]= => word[LONGTAG1]TAG2]=

DEFTYP
1: 32

CHARSET
Ltr: 48-57 65-90 97-122
Maj: 65-90

STATESET
Cut: 10

RULES     LC   RC    SC   RS    MV
[; [;     Ltr  Ltr    1    2   -5
]+[; [;  Ltr  Ltr    1    2   -5
]; ;     Ltr  Ltr    2   1   5

cipl.rul

_Numeral--+morph[TAG] => Numeral-morph[TAG]
_"Word"+-morph[TAG] => "Word"-morph[TAG]

DEFTYP
1: 32

CHARSET
Ltr: 48-57 65-90 97-122"

STATESET
Cut: 10
### oms16.rul

<table>
<thead>
<tr>
<th>RULES</th>
<th>LC</th>
<th>RC</th>
<th>SC</th>
<th>RS</th>
<th>MV</th>
</tr>
</thead>
<tbody>
<tr>
<td>_word[TAG] =&gt; word/Tagg</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>_word-[TAG] =&gt; word-/Tagg</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### CHARSET

Ltr: a-z A-Z 48-57 " - 

### oms18.rul * 4

@set in _ between the tags

### CHARSET

Ltr: a-z A-Z 48-57 " - 

### RULES

<table>
<thead>
<tr>
<th>RULES</th>
<th>LC</th>
<th>RC</th>
<th>SC</th>
<th>RS</th>
<th>MV</th>
</tr>
</thead>
<tbody>
<tr>
<td>/; /;</td>
<td>0 0 1 2 5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DET; _DET;</td>
<td>- 0 2 3 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FN; _FN;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HA; _HA;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IGE; _IGE;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IK; _IK;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>INF; _INF;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ISZ; _ISZ;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>KOT; _KOT;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>KSZ; _KSZ;</td>
<td>Ltr 0 2 3 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MN; _MN;</td>
<td>- 0 2 3 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NM; _NM;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NU; _NU;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NUM; _NUM;</td>
<td>Ltr 0 2 3 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SZN; _SZN;</td>
<td>- 0 2 3 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>USZ; _USZ;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>e1; _e1;</td>
<td>E 0 2 3 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>e2; _e2;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>e3; _e3;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t1; _t1;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t2; _t2;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t3; _t3;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Te1; _Te1;</td>
<td>- 0 2 3 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Te2; _Te2;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Te3; _Te3;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tt1; _Tt1;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tt2; _Tt2;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tt3; _Tt3;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Me1; _Me1;</td>
<td>E 0 2 3 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Me2; _Me2;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Me3; _Me3;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mt1; _Mt1;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mt2; _Mt2;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mt3; _Mt3;</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
TMe1; _TMe1; 0 2 3 3
TMe2; _TMe2;
TMe3; _TMe3;
TMt1; _TMt1;
TMt2; _TMt2;
TMt3; _TMt3;
Fe1; _Fe1; E 0 2 3 3
Fe2; _Fe2;
Fe3; _Fe3;
Ft1; _Ft1;
Ft2; _Ft2;
Ft3; _Ft3;
Pe1; _Pe1;
Pe2; _Pe2;
Pe3; _Pe3;
Pt1; _Pt1;
Pt2; _Pt2;
Pt3; _Pt3;
TFe1; _TFe1; 0 2 3 3
TFe2; _TFe2;
TFe3; _TFe3;
TFt1; _TFt1;
TFt2; _TFt2;
TFt3; _TFt3;
TPe1; _TPe1;
TPe2; _TPe2;
TPe3; _TPe3;
TPt1; _TPt1;
TPt2; _TPt2;
TPt3; _TPt3;
Ie1; _Ie1;
IMe1; _IMe1;
IFe1; _IFe1;
IPe1; _IPe1;
INRe1; _INRe1;
INRe2; _INRe2;
INRe3; _INRe3;
INRt1; _INRt1;
INRt2; _INRt2;
PSe1; _PSe1;
PSe2; _PSe2;
PSe3; _PSe3;
PSt1; _PSt1;
PSt2; _PSt2;
PSt3; _PSt3;
PSe1i; _PSe1i;
PSe2i; _PSe2i;
PSe3i; _PSe3i;
PSt1i; _PSt1i;
PSt2i; _PSt2i;
PSt3i; _PSt3i;
DIS; _DIS;
ESS; _ESS;
IKEP; _IKEP;
KIEM; _KIEM;
MOD; _MOD;
MUL; _MUL;
SKEP; _SKEP;
PL; _PL;
POS; _POS;
FF; _FF;
ABL; _ABL;
ACC; _ACC;
ADE; _ADE;
ALL; _ALL;
CAU; _CAU;
DAT; _DAT;
DEL; _DEL;
ELA; _ELA;
FAC; _FAC;
INE; _INE;
ILL; _ILL;
INS; _INS;
FOK; _FOK;
FOR; _FOR;
NOM; _NOM;
SOC; _SOC;
SUB; _SUB;
SUP; _SUP;
TEM; _TEM;
TER; _TER;

clip18.rul
@ /_Tagg => /Tagg

RULES     LC   RC   SC   RS   MV
/__; /;     0    0    1    2    5

menrad.rul
_one sentence per line

PARAM
TOT=1000

DEFTYP
1: @
3: 31

oms17.rul
_delete @ in the end of the line

PARAM
TOT=1000

DEFTYP
1: @
3: 31

RULES     LC   RC   SC   RS   MV
#@#; ;     0    #    1    2    5
NORMHUT.BAT
_delete all subtags (Word/Tag1_Tag2_TagN => Word/Tag1)

@ECHO OFF
pcbeta huvud.rul oms16ut.txt huvudut.txt
pcbeta menrad.rul huvudut.txt huvmenr.txt
pcbeta oms17.rul huvmenr.txt normhut.txt
del huvudut.txt
del huvmenr.txt

huvud.rul
_delete all subtags
_input: oms16ut.txt
_output: huvud.txt

DEFTYP
1: 32

CHARSET
Ltr: 48-57 65-90 97-122
Del: /

STATESET
Cut: 10

RULES    LC  RC  SC  RS  MV
FN; FN;   Del Ltr 1 2 -5
USZ; USZ;
MN; MN;
HA; HA;
IGE; IGE;
DET; DET;
MSZ; MSZ;
MOD; MOD;
KSZ; KSZ;
KOT; KOT;
IK; IK;
NM; NM;
SZN; SZN;
ISZ; ISZ;
INF; INF;
ROV; ROV;
MIN; MIN;
HI; HI;
NU; NU;
NUM; NUM;
AUY; AUY;
!; !;
?; ?;
.; .;
....; ...;
; ;
.; .;
-; -;
*; *;
menrad.rul
_one sentence per line

PARAM
TOT=1000

DEFTYP
1: @
3: 31

oms17.rul
_delete @ in the end of the line

PARAM
TOT=1000

DEFTYP
1: @
3: 31

RULES    LC  RC  SC  RS  MV
@#; ;    0   #   1   2   5
Normalisation of the test (‘hand’) corpus

NORHAND.BAT
_change the input text to the form ‘Word/Tag1_Tag2_TagN’

@ECHO OFF

pcbeta omv1.rul sagain.txt omv1ut.txt
pcbeta omv2.rul omv1ut.txt omv2ut.txt
pcbeta omv3.rul omv2ut.txt omv3ut.txt
pcbeta omv4.rul omv3ut.txt omv4ut.txt
pcbeta omv5.rul omv4ut.txt omv5ut.txt
pcbeta tomrad.rul omv5ut.txt tomraut.txt
pcbeta omv6.rul tomraut.txt omv6ut.txt
pcbeta omv7.rul omv6ut.txt omv7ut.txt
pcbeta omv8.rul omv7ut.txt omv8ut.txt
pcbeta omv9.rul omv8ut.txt omv91ut.txt
pcbeta omv9.rul omv92ut.txt omv9ut.txt
pcbeta omv10.rul omv9ut.txt omv10ut.txt
pcbeta omv11.rul omv10ut.txt sagaut.txt
del omv1ut.txt
del omv2ut.txt
del omv3ut.txt
del omv4ut.txt
del omv5ut.txt
del tomraut.txt
del omv6ut.txt
del omv7ut.txt
del omv8ut.txt
del omv91ut.txt
del omv92ut.txt
del omv9ut.txt

_omv1.rul
#%Word => Word\[USZ]

CHARSET
Ltr: 48-57 65-90 97-122

RULES       LC  RC  SC  RS  MV
#%; ;       #   Ltr  1   2   5
#; \[USZ];  Ltr  #   2   1   5

_omv2.rul
_1 PUNCT  mark  mark\TAG => mark\TAG
_the rule does not include the ; hence,
_1 PUNCT ; ;\TAG => ; ;\TAG

CHARSET
Del: 32-47 58-64 91-96
Ltr: 65-90 97-122
Tag: \

115
STATESET
Cut: 10

RULES
#1  PUNCT  ; #; #  Del  1  2 -5
   ; ;    0  Tag  2 10 -4
   ; ;
   ; ;
   ; ;
   ; ;
   ; ;
   ; ;
   ; ;

_omv3.rul
_TOK  word  
_token\[TAG\] => word\[TAG\]

CHARSET
Ltr: 48-57 65-90 97-122

STATESET
Cut: 10

RULES
#1  TOK  ; #;   #  Ltr  1  2 -3
   ; ;   Ltr  Ltr  2  3 -5
   \; \;        Ltr \[    3  10  4

_omv4.rul
_Word  BOS  Token\[TAG\] => Word\[TAG\]

CHARSET
Ltr: 48-57 65-90 97-122
Maj: 65-90

STATESET
Cut: 10

RULES
BOS ; ;  Ltr Maj 1 2 -3
EOS ; ;  Ltr Maj 1 2 -3
\; \;        Ltr \[    2  10  4

_omv5.rul
_word_wrong§wordcorrect\[TAG\] => wordcorrect\[TAG\]
_WordWrong§WordRight\[TAG\]

CHARSET
Ltr: 48-57 65-90 97-122
Maj: 65-90

STATESET
Cut: 10

RULES
#; #;  #  Ltr  1  2 -3
$; ;    0  Ltr  2 10  5

116
_tomrad.rul
_eliminate lines containing 
_or 

 CHARSET
 Siff: 0 1 2 3 4 5 6 7 8 9 *

 RULES     LC   RC  SC   RS   MV
 #/;;       #    #   1    2    0
 ##/;;
 #/**;;      #   Siff 1    2    0

_omv6.rul
_mark\XPUNCT => mark

 RULES     LC RC SC RS
 \WPUNCT; ; 0  #  1 2
 \SPUNCT;

_omv7.rul
_tag all punctuation mark

 RULES     LC   RC      SC    RS
 ,; ,/;;     #     #    1     2
 ;; :/;;     #     #    1     2
 (; /(;       #     #    1     2
 .; ./.@;     #     #    1     2
 ?; ?/?@;     #     #    1     2
 !; !/?@;     #     #    1     2
 ); )/)@;     #     #    1     2
 ...; .../...@;  #     #    1     2
 #--; -/--;     #     #    1     2

_omv8.rul
_delete + and all empty lines

 RULES  LC RC SC RS  MV
 +; ; 0  0  1  2  5
 #;;     # #  1  2  0

_omv9.rul
£][ => _

 RULES     LC   RC  SC   RS   MV
 ][; _; 0  0  1  2  5

_omv10.rul
_delete [] and change \ with /

 RULES     LC   RC  SC   RS   MV
 \[]/; 0  0  1  2  3
 ]; ; 0  0  2  1  5
_omv11.rul
_one sentence per line

PARAM
TOT=1000

DEFTYP
1: @
3: 31

RULES    LC  RC  SC  RS  MV
@#; ;     0   #   1   2   5

SHEADTAG.BAT
_delete the subtags

@ECHO OFF

pcbeta headtag.rul omv10ut.txt headut.txt
pcbeta omv11.rul headut.txt sheadta.txt
del headut.txt

__headtag.rul
_delete the morphological tags
_input files: omv10ut.txt and dikttag.txt
_word/Tag1_Tag2_TagN => word/Tag1

CHARSET
Ltr: 48-57 65-90 97-122 1 2 3
Del: /

DEFTYP
3: 1-255

STATESET
Cut: 10

RULES    LC  RC  SC  RS  MV
USZ_; USZ;   Del  Ltr  1  2  -5
FN_; FN;
MN_; MN;
HA_; HA;
IGE_; IGE;
DET_; DET;
MSZ_; MSZ;
MOD_; MOD;
KSZ_; KSZ;
KOT_; KOT;
IK_; IK;
NM_; NM;
SZN_; SZN;
ISZ_; ISZ;
INF_; INF;
ROV_; ROV;
MIN_; MIN;
Hi_; Hi;
Nu_; Nu;
Num_; Num;
Aux_; Aux;
#; #;        Ltr    #     2     10    5

_omv11.rul
_one sentence per line

PARAM
TOT=1000

DEFTYP
1: @
3: 31

RULES    LC  RC  SC  RS  MV
@#; ;     0   #   1   2   5

SDELTAG.BAT
_delete tags and put one sentence per line

@ECHO OFF

pcbeta keep@.rul omv10ut.txt deltagu.txt
pcbeta menrad.rul deltagu.txt sagadel.txt
del deltagu.txt

_keeprul.rul
_delete tags but keep @

STATESET
Cut: 10

RULES    LC  RC  SC  RS  MV
/; ;    0   0   1   2  -3
@; @;    0   #   2   10  4
#; ;    -@   #   2   10   5

_menrad.rul
_one sentence per line and delete @

DEFTYP
1: @
3: 31

RULES    LC  RC  SC  RS  MV
@#; ;    0   #   1   2   5
Appendix E: Output

In texts with only PoS tags, the items with wrong part of speech tags are underlined. In texts with PoS and subtags (i.e. tags which provide information about inflectional properties of a token) wrongly annotated tokens are underlined. If there is more than one tag attached to a token then the wrongly annotated tag is printed in bold. Items which are correctly annotated but missing one or several subtags are italicized.

**Texts with PoS tag**

Poem: I. Ágh: Az én erdőEm

Kő2zel/MN hú1zőldik/IGE el1elemhez/IGE ./, s/KOT nem/HA tudom/IGE ./, mekkora/FN ./, holo/HA vél1gző3dík/IGE ./, fal/FN nekem/NM o3/NM ./, el1o3/MN panasz/falom/FN ./.
A/DET ritkul1o1/FN leveles/MDN a1thull01/FN fe1lNy/FRN elke1pzelem/FRN ./, a/DET to2lgy/FRN ./, bu2kk/FN ./, szőlő leveleke/FRN fődrozta/IGE fe1lNy/FRN ./, az/DET esz03u1n/FRN pocsolya1kh0z/INF hasonl01/MN ayart/IGE ./, vadleso3/MN magaslatai/FRN ./, amelyek/NM tekinettem/FRN f02lvezet1k/IGE az/DET el1g/FN f1tylaira/FRN ./.
Lzgat/FRN a/DET megholdott/FRN rejtő1yu3/MN szal1nt0k/FRN uta1n/NU ./, szeretne1k/IGE oda/futni/INF a/DET to2bbieko3t0l/MN ./, szeretne1k/IGE meglesni/INF egy/DET vadat/FRN ./, hajtani/INF ./, elveszteni/INF els/KOT u1jra/HA megtala1l1n/INF ./.
Szeretne1k/IGE elveszni/INF els/KOT si1rni/INF ./, el03keru2ln/INF ./, ral1hajolni/INF a/DET korareggel/MN tu3zhele/FRN mele1ge1r1/IGE a/DET hazate1r1/FRN hadifogyok/FRN o2r02me1vel/FRN ./.
Vonz/FRN ./, mert/KOT titkait/FRN sohasem/HA fedte/IGE f02l/HA elo3ttem/IGE els/KOT sohasem/HA hagyta/IGE ./, hogy/KOT titkait/FRN megtala1l1jami/FRN ./.
A/DET bozo1tho1l/FRN kiugro1/FRN szarvast/FRN ki1l/IK la1tta/IGE ?/?
Igaz/MN ./, olyan/NM mint/KOT a/DET szappanreka1mon/FRN ./, els03/FRN szan1ba1i/FRN maga/NM ala1NU hul1zva/HA ./, szarvai/MDN el1ge/FRN vetve/HA sz02kell/FRN ./, repu2l/IGE olyan/NM f02nsel1gesen/MDN ./, hogy/KOT lelo3ve/HA is/KOT u1gy/FN bukik/IGE ./, ahogy/Ha/a/DET szabadsa1lg/FRN nagy/NM vadja1nak/FRN bukin/INF illik/IGE ./.
La1ttam/IGE a/DET szarvast/FRN rabnak/FRN ./, fiatal/NM els/KOT megtel1palzott/MDN volt/IGE ./, mint/KOT akit/MDN e1lve/HA fognak/IGE el/IK ./, mint/KOT akit/MDN ero3szakos/NM kezekkel/FRN ./, te1rdekel/FRN magu1ke1val/FRN teszenek/IGE ./.
Nem/HA la1ttam/IGE szabadn/IGE ./.
Ha/KOT nem/HA lettem/IGE olyan/NM szabadnufot01/MN ./, szabadon/MDN do3zso2lo3/FRN ./, szabanon/MDN szenved03/FRN ./, hali/HA meglal1tni/FRN a1hi1toztam/INF legal1bb/HA ./.
Mert/KOT csak/HA a/DET gy1alva/NU nyu1lat/FRN ./, a/DET sompolygo1/MN rol1ka1t/FRN ./, a/DET szel1szalado1/MDN égereket/FRN ismerem/IGE ./.
Puskal1val/FRN menyek/IGE oda/IK ?/?
Ma1r/HA csak/KOT fegyverrel/FRN juthatok/IGE oda/IK ./, ahova/KOT valamikor/HA u2res/NM ke1zzel/FRN akartam/IGE ?/?
Ilgy/KOT nyerje1k/FRN biztona1got/IGE ?/?
Hu1zzon/FRN ./, vezessen/NM ./, leskelo3dteszen/MN ./, lo2vessen/MDN ?/?
Borral/FRN els/KOT friss/NU vadu1sall/FRN megki1nallasson/IGE ?/?
Engedjen/FRN bo3jegel1s/FRN hallami/INF ./, la1tmi/INF hajnali/INF forra1st/FRN ./, megtudni/INF titkot/FRN ?/?
Nem/HA kell/IGE nekem/NM a/DET kutatásbo1l/FN passzió1/FN ./, veszéllybo3l/MN seltaligatás/FN !/
Vagy/KOT nincs/IGE ma1r/HA ez/NM az/DET erdő3/FN ??
Vagy/KOT talán/HA benne/HA vagyok/IGE ??
S/KOT mindig/HA egy másra/NM gondolunk/IGE ./, amint/KOT ja1rom/SZN o2svélynyei/FN egyedű2l/HA ./, u2res/MN ke1zzel/FN ./.
Lendu2l/FN a/DET hab/FN s/KOT a/DET part/FN falán/FN ezut2stsarkányul1s/MN la1ba/FN dobban/FN :: ne1zd/IGE ./, fu2rdik/IGE a/DET fekete/MD la1ny/FN ./, fekete/MD la1ny/FN fehe1r/MN habokban/FN ./.
Elszenderu2lt/FN a/DET bul/MN szeme1n/FN ./, hullalma1ba/FN hull/IGE ma/HA teste/FN ./, lelke/FN ./, hullalmo1s/IGE hajú1/MN vo3lege1ny/FN milyen/NM ero3sen/MN a1t02lele1te/IGE ./.
De/KOT ne1zd/IGE :: so2te1t/MN erdő3/FN ko2zo2tt/NU nagy/NM ./, ordas/MN fellegek/FN szakadnak/IGE S/KOT jo2n/FN a1rjja/MN za1lpor02tt/FF go2rge1teseges/IGE hegyi1/IGE patakna1k/NU ./.
Haragos/MN ./, szennes/NM a1r1adat/FN ./, a/DET medre1t/FN karmok/FN a1sza1k/IGE ./.
Ke1rd/IGE meg/IK a/DET ho2kkent/FN ga1talakat/IGE ./, ga1ncsolja1k/IGE a/DET vizek/FN futa1sal1t/FN ./.
Ne/HA essen/MN folt/FN fe1nyes/NM haja1n/FN ./, iszapos/MN a1r/FN ho2zza1/IGE ne/Ha e1rjjen/FN ./, fekete/MD la1ny/FN falán/FN ha1/IGE ./, talán/HA utolszov/KOT fu2rdik/IGE ./- ho1fehe1ren/FN ./.

Fairy tale: B. Balázs: A három hísses királyleány

BALA1ZS/FN BE1LA/FN :: A/DET HA1ROM/FN HU3SE1GES/FN KIRA1LYLEA1NY/FN/FN
El1t/IGE egyszer/HA Kandrapura/IGE val1rosa1ban/FN egy/DET Suryakanta/FN nevu3/MN hatalmas/MN király1/FN ./, aki/NM haldrom/SZN fo3tulajdonsa1ga/FN a1ltal/NM olyan/NM volt/IGE a/DET fe1r0fiak/FN ko2zo2tt/NU mint/KOT a/DET ko2vek/FN ko2z0t/FN az/DET Adamanth/FN ./, tu2ndo2kl03/MN ./, tiszta/MN e1s/KOT keme1ny/IGE ./.
Tu2ndo2kl03/MN szel1pse1ge/FN minden/NM asszonyt/FN szi1ven/FN harapott/MN MN mint/KOT a/DET kobra/FN ./.
Bu3ntelen/MN folt/IGE fe1nyes/NM haja1n/FN ./, iszapos/MN a1r/FN ho2zza1/IGE ne/Ha e1rjjen/FN ./, fekete/MD la1ny/FN falán/FN ha1/IGE ./, talán/HA utolszov/KOT fu2rk1t/IGE ./- ho1fehe1ren/FN ./.

Suryakanta/FN király1/FN csak/HA hullatta/IGE o3ket/NM ./, de/KOT rajtuk/NM nem/Ha val1tozatot1/IGE ./.
Suryakanta/FN király1/FN uta1lta/FN a/DET szavakat/FN e1s/KOT minden/NM keme1ny1/FN e1s1zt/IGE ./, de/KOT rajtuk/NM nem/Ha val1tozatot1/IGE ./.

Suryakanta/FN király1/FN uta1lta/FN a/DET szavakat/FN e1s/KOT minden/NM keme1ny1/FN e1s1zt/IGE ./, de/KOT rajtuk/NM nem/Ha val1tozatot1/IGE ./.
Suryakanta/FN király1/FN uta1lta/FN a/DET szavakat/FN e1s/KOT minden/NM keme1ny1/FN e1s1zt/IGE ./, de/KOT rajtuk/NM nem/Ha val1tozatot1/IGE ./.
Mert/KOT Balapandita/FN sze1pse1gelben/IGE a/DET hold/FN go2mbo2lyu3se1ge/FN ./, fu3/FN remegelse/FN ./, leve1/FN ko2nnyu3se1ge/FN ./, elefa1ntorma1ny/MN ve1konyula1sa/FN ./, melhek/MN rajza1sa/FN e1s/KOT a/DET papaga1j/FN melle/FN la1gysa1ga/FN egyesu2tek/IGE mint/KOT egyma1st/NM relg/MN keros3/MN szerelmesek/FN ./.

-/- Mie1lt/NM hi1vat/IGE uram/FN ./, Suryakanta/MN kira1ly/FN -/- ke1rdezte/IGE Balapandita/IGE ./.

De/KOT Suryakanta/FN nem/HA felelt/IGE hanem/KOT odale1pett/IGE a/DET sze1pse1gelben/IGE homlokon/FN cso1kolta/IGE olyan/NM jeleket/FN te1ve/HAn/HA mint/KOT a/DET bu1csu1zol1k/FN ./.

-/- Uram/IGE Suryakanta/FN -/- szo1lt/IGE Balapandita/IGE akkor/HA -/- homlokon/FN csol1kolsz/IGE e1s/KOT olyan/NM jeleket/FN mint/KOT a/DET bu1csu1zol1k/FN ./.

Vajon/KOT bu1csu1zni/INF akarsz/IGE to3lem/NM te/NM fel1rfitigris/FN ./? Suryakanta/FN nem/HA felelt/IGE hanem/KOT odale1pett/IGE a/DET ifju1sa1ga1nak/FN ta1rsa1hoz/FN fordult/IGE ./, mondva1n/FN :/- -/- Razakosa/FN ke1szu2lj/FN u1tra/IGE ./.

De/KOT ki1se1ret/FN ne/HA ko2vessen/MN e1s/KOT fullaja1rok/MN elo3tu2nk/IGE ne/HA fussenak/IGE ./.

E/NM szavako10l/FN Balapandita/IGE e1s/KOT Ragazakosa/FN mege1rtette1k/IGE hogy/KOT Suryakanta/IGE kira1ly/FN az/DET erdo3be/FN indul/IGE valamegy/IGE szent/MN liget/MN la1togata1sa1ra/FN ./, hogy/KOT a1ldozatala/FN ./, fu2rde1ssel/FN e1s/KOT medita1cio1val/FN tiszti1tsa/IGE lelke1t/FN ./.

Akkor/HA Balapandita/FN hangja1ra/IGE nedves/MN fa/FN illata1tol/FN megdu2ho2tt/IGE pe1zsmaszarvasok/Tu1rja1k/IGE magukat/NM a/DET sziromhullato1/MN ja1zminbokrokon/FN keresztu2l/HA ./, Suryakanta/HA kira1ly/FN hirtelen/MN mega1llt/IGE ./.

-/- Mit/NM la1ltta1l/IGE uram/FN ?/? -/- ke1rdezte/IGE Razakosa/IGE ./.

Suryakanta/FN akkor/HA lehajolt/IGE e1s/KOT mikor/NM megint/HA kii/IK ./ Azuta1n/IGE mind/NM a/DET kett03t/SZN eldobta/IGE e1s/KOT tovla1bb/IK akart/IGE menni/INF ./.

Hanem/FN akkor/HAn/HA a/DET fekete/MN ki1gyol/FN felagaskodott/IGE elo3tte/NM elo3t/NM ta1rsu2lta1nak/FN ta1rsu2lta1/MN ./.

Mikor/NM ma1r/HA me1llyen/MN ja1rtak/IGE az/DET erdo3ben/FN ahol/HA a/DET Kadamba/FN fa/FN illata1l1/FN megu2o2do2u2t/IGE pelzsmaszarvasok/FN tu1lja1/FK/IGE magukat/NM a/DET sziomhullato1/MN ja1zminkokrokon/FN keresztu2l2/FN ./, Suryakanta/HA kira1ly/FN hirtelen/MN megia1lt/IGE ./.

-/- Mit/NM la1tta1l/IGE uralam/FN ?/? -/- ke1rdezte/IGE Razakosa/IGE ./.

Suryakanta/FN kira1ly/FN akkor/HA lehajolt/IGE e1s/KOT mikor/NM megint/HA kiegyenesedett/IGE jobb/MN kezelve1/FN egy/DET nagy/MN fekete/MN kilgyol/FN torka1t/FN fogta/IGE ./, bal/MN kezelve1/FN pedig/KOT egy/DET kis/IGE MN nyulat/FN ./, melyet/FN a/DET kilgyol1/MN szalja1bo1/FN vett/IGE ik/IK ./.

Azuta1n/IGE mind/NM a/DET ketto3t/SZN eldobta/IGE e1s/KOT tova1bb/IK akart/IGE menni/INF ./.

Hanem/FN akkor/HAn/HA a/DET fekete/MN kilgyol/FN felagaskodott/IGE elo3tte/NM elo3t/NM ta1rsu2lta1nak/FN ta1rsu2lta1/MN ./.

Mie1lt/NM hi1vat/IGE uram/FN ./, Suryakanta/MN kira1ly/FN ?/? ./.
Mert/KOT tudt/IGE meg/KOT o1jh/FN Suryakanta/FN kira1ly/FN, hogy/KOT harminc/SZN napig/FN nem/HA ettem/IGE e1s/KOT a/DET hala1l/FN sarka/FN van/IGE fejemen/FN. 

Tudsz/IGE e-NM nekem/NM ma1s/NM talpla1le1kot/FN adni/INF a/DET nyu1l/FN helyett/NU ?/?
Suryakanta/FN kira1ly/FN mozdulatlan/MN arccal/FN ne1lzte/IGE a/DET fekete/IN de1s/KOT nem/HA felet/IGE, hanem/KOT kihu1lta/IGE to3re1t/IGE e1s/KOT izmos/MN barna/MN karja1bo1l/FN kanyar1lott/IGE egy/DAT darabot/FN e1s/KOT azt/NM odadobta/IGE a/DET ki1gyo1l/MN ele1/NU. 

Hanem/FN a/DET ki1gyo1l/HA nem/HA nyu1l/IGE uta1na/NU. 

-/- Meghalt/IGE -/- mondta/IGE Razakosa/IGE. 

-/- Nem/HA mondad/IGE, hogy/KOT segi1teni/INF akarsz/IKE rajta/IK, hal1t/HA eleresztette/IGE e1llete1t/FN mert/KOT nem/HA tudta/IGE mire/HA val1jron/IGE mel1g/HA toval1bb/IK szenyedve/IGE. 
Suryakanta/FN kira1ly/FN mozdulatlan/MN arccal/FN ne1lzte/IGE a/DET hosszu1l/MN, fekte/MN hulla1t/FN. 

Le1k1l/IGE oly/NM keve1ssel1/NM illette/IGE a/DET halal1/FN mint/KOT a/DET vad/MN elefa1ntot/FN az/DAT erde1/NU/me1l/NM ra1ncos/MN hal1ta1ra/FN hull/IGE. 

El1ppen/HA a1l/IKE akart/IKE le1pni/FN rajta/NM, hogy/KOT toval1bbmenjen/FN mikor/FN hirtelen/MN megjelent/IGE elo3tte/NM Ganesa/IGE az/DAT elefa1ntarcu1/MN isten/FN. 

-/- Az/DAT istenek/FN o2ltek/IGE akik/NM terme1szetemet/FN u1gy/HA termelte1k/IGE hogy/KOT tu2zes/MN korallbokr1ko1t1nt/FN pirosloktatt/IGE benne/NM az/DAT erkek/FN. 
Ro1lzaszi1n/MN suga1r/MN orma1lnya1t/FN tal1ncban/FN emelte/IGE e1s/KOT ilgy/KOT szo1lt/IGE :/: -/- Suryakanta/HAla2le1/IWE. 

Suryakanta/FN mozdulatlan/MN szemmel/FN ne1lzett/IGE Ganesa/IGE kis/MN szeme1be/IGE e1s/KOT ilgy/IKE felet/IGE. 

-/- Az/DAT istenek/FN o2ltek/IKE aik/NM termel1szemet/FN emelte/IKE hogy/KOT tu2zes/MN korallbokrok/IKE arctal1nt/FN IGE benne/NM az/DAT erkek/FN. 

Le1kem/FN vir1lga/FN elo3bbi/MN e1lletekben/FN gyro2keredzik/IGE e1s/KOT sem/KOT mozdulni/INF sem/KOT val1ltozni/INF nem/HA tud/IGE. 

Akkor/NM Ganesa/FN nevetett/IGE :/: -/- Ha/KOT lelk1d/FN sem/KOT mozdulni/INF sem/KOT val1ltozni/INF nem/HA tud/IGE, hal1t/HA majd/HAla1t/le1l/FN szai1ra1rol1l/FN, hogy/KOT mozduljon/IGE e1s/KOT keme1lnyse1g/IGE F/N miatt/NM to2bbe1l/HAla gyilkos1gb1a/FN ne/HA es1ekl/IWE. 

Ezt/NM mondta/IGE nevetve/IGE Ganesa/FN e1s/KOT ro1lzaszi1n/FN hosszu1l/MN suga1r/MN orma1lnya1val/FN belenyu1lt/IWE Suryakanta/FN melle1be/IGE e1s/KOT lel1ke/FN vir1lga1t/FN lele1pte/IWE egyszer1ro1l/IWE. 

Avval/FN eltu3nt/IWE. 

A/DET kira1ly/FN melle1hez/FN kapott/IGE mintha/KOT nyu1l/FN fu1rta/IKE volna/IKE a1l/e1s/KOT felka1lott/IGE :/: -/- O1jh/FN jaj/FN nekem/NM !/!

Akkor/NM Razakosa/FN kiejtette/IGE keze1bo3l/IWE i1ja1t/FN., mert/KOT soha/HA senki/FN mel1g/HA Suryakanta/IKE kira1ly/FN jajszava1t/IWE nem/HA hallotta/IGE. 

-/- Uram/IKE Suryakanta/FN -/- k1rdezte/IGE halkan/MN mint/KOT aki/NM a/DET fe1lelto3l/FN fe1l/e1s/KOT a1llomnak/FN szeret1el/IGE hinni/INF o2nmaga1t/NM. 

-/- Uram/IGE. -/- mi/NM baj/FN e1rt/IWE?/!

Suryakanta/FN i1jedten/FN ford1lta1t/IGE feje1t/IGE Razakosa/FN fele1/NU e1s/KOT mel1yen/MN elpir1ult/IGE. 

-/- Oh/NNP kedves/MN Razakosa/FN -/- mondta/IGE szel1gyenszen/IGE mosolyoga/HAla. 

-/- Nagyon/HAla fa1j/IKE sebe1m/IGE., melyet/FN ama/NM k1lygo1l/MN miatt/NM a/DET karomba/FN val1g1tam/IGE. 

Akkor/NM Razakosa/FN kiejtette/IGE keze1bo3l/IWE tege1z1t/FN e1s/KOT szeme1/FN ne1ma1n/FN lesu2to2tte/IGE mert/KOT Suryakanta/IGE olyat/NM mondott/IGE.,
amit/NM egy/DET ko2zo2nse1ges/MN Ksatria1nak/FN sem/KOT szabad/MN ,/, nemhogy/MN kira1lynak/FN ./.
-/- Elejtetted/IGE i1jad/IGE e1s/KOT tegezed/IGE ./, kedves/MN Razakosa/FN -/- mondta/IGE Suryakanta/IGE e1s/KOT felemelte/IGE sietse1ggel/IGE mint/KOT valami/NM hi1zelo3/FN szolga/FN ./.
-/- Mie1r/HA su2r/IGE le/IK a/DET szemed/FN Razakosa/FN e1s/KOT mie1rt/HA salpadsz/IGE el/IK ??
Tal1n/HA bizony/HA raitam/FN la1tsz/IGE valami/NM va1ltoza1st/FN ??
Oh/NNP milyn/EN va1ltoza1s/FN i1nye1t/IFC eshetne1k/FN rajtam/FN ?/!
Suryakanta/FLN szapora/FLN szavainak/FLN cso2rge1se/FN ulgy/HA hullott/IGE Razakosa/IGE fu2le1be/FLN ./, mint/KOT egy/DET o2sszo2r/EN dra1ga/MN ede1ny/FLN cserepeinek/FLN cso2ro2mpo2le1se/FLN ./.
-/- Legyu2nk/FLN vido1mak/IGE -/- mondta/IGE Suryakanta/IGE -/- e1s/KOT sije1snu2nk/FLN fe1se1lgemhez/FN Balapandita1hoz/IGE ./, mert/KOT szemem/FN e1s/KOT karam/FLN ulgy/HA ki1valnia1l/FLN Balapandita1t/IGE ./, hogy/KOT leva1lnak/FLN testermo3l/IGE e1s/KOT hozza1l/IFC repu2lnak/IGE ha/KOT nem/HA ha siete1k/IGE ./.
-/- Uram/IGE Suryakanta/IGE kiraly1/FN -/- mondta/IGE Razakosa/IGE -/- a/DET szen1t/IFC MN liget/MN la1toga1salra/FN indultunk/FN ./, hogy/KOT a1ldozzunk/IGE az/DET isteneknek/FLN ./.
Suryakanta/FLN elva1ltozott/IGE arca/IGE mint/KOT az/DET ijedt/IFC gyermeke1/IGE olyan/NM lett/IGE ./.
-/- Igazal1n/HA aze1r/KOT indultunk/IGE ??
Jo1/MN ./, nem/HA ba1nom/IGE ./.
Akkor/FLN ni1ssemi2nk/IGE a/DET szen1t/IFC MN ligetbe/FLN ./.
Alik/HA haladatk/IGE nei1halny/EN le1pel1st/FN Suryakanta/FLN kiraly1/FN hi1rtem/EN MN visszafordult/IGE e1s/KOT i1gyy/HA KOT szo1l1t/IFC ./.
-/- A/DET bu2szke/MN Balapandita/FLN arcke1pem/IGE elo3tt/NU u2l/IGE e1s/KOT va1r/KOT ./.
Gondolatai/FLN la1bam/IFC ko2re1/NU hurkolo1dnak/IGE e1s/KOT nem/HA eszterne1k/IFC tova1bb/IK ./.
Kedv1n/HA Razakosa/FLN ke1rle1k/IGE ez/EN egyszer/HA te1rju2nk/IFC inka1bb/HA ha siete1k/IGE tova1bb/IK ./.
Akkor/IFC Razakosa/FLN hangos/MN si1ral1ssal/IFC feljajdult/IGE ./,/- O1h/IFC Suryakanta/FLN ./, Suryakanta/HA mi/NM lett/IGE belo3l1d/NU !/!
Suryakanta/FLN elva1ltozott/IGE szeme/FN hunyorgatva/IFC ne1zett/IGE mint/KOT egy/DET gyalva/IFC MN kolduse1/FN ./.
-/- Mie1r/NNM ja1gat/IGE Razakosa/FLN ?/!
Tal1n/HA valami/NM va1ltoza1s/FN esett/IFC raitam/IFC ?/!
Mife1le/IFC MN va1ltoza1s/FN eshetne1k/IFC rajtam/IFC ./, holott/IGE e1r/NM vagyok/IFC Suryakanta/FLN kiraly1/FN ./, fe1r1fiak/IFC ko2z1t/NU olyan/NM mint/KOT a/DET ko2vek/FLN ko2z1t/NU az/DET adamanti/IFC ./.
E1n/NM vagyok/IFC a/DET mozdni1hatatalan/MN Suryakanta/FLN ./.
Tova1bb/IK ./, nem/HA besze1lt/IGE a/DET kiraly1/FN hanem/KOT arca/IFC e1l1/NU csapta/IGE kezeti/IFC MN e1s/KOT hungasan/MN felzokogott/IGE ./,/- Oh/NNP Razakosa/IFC ifju1sal1gornak/IFC talrsa1/MN ./, nem/HA vagyok/IFC e1n/NM vagyok/IFC Suryakanta/FLN ko2nnyei1/FN elo3hi1vta1l/IFC testve1re1tek/IFC mint/KOT ahogy/HA egyszer/HA egy/DET felbukkano1/MN hangya/IFC elo3hi1vja1/FN mind/NM a/DET to2b1it/IFC MN ./,
-/- Uram/IFC ez/EN MNi1nas/FN bu2nteta1l/IFC a/DET ki1gyo1le1r/IFC ./.
Lete1pte1/IFC lelked/IFC virag1al1/FN szal1ral1ro1l/IFC e1s/KOT most/HN mint/KOT szakaszto1tt/IFC lo1lnusz/IFC hulla1mos/MN tanov/FN inog/IGE e1s/KOT halnyko1dik/IFC ./, re1szeg/MN ta1ncoske1nt/IFC ./, bizonytalanul/MN ./,
-/- Taj/ISZ ./, jaj/MN Razakosa/MN segi1ts/FN raitam/IGE ./, kedves/MN Razakosa/FN ?/!

Fogd/IGE meg/IK lelkemnek/IGE lelke1t/FN , hogy/KOT megint/HA a1llando1/MN legyen/IGE ./.
Szo1li1tsd/IGE neve1n/FN , hogy/KOT horgonya/FN legyen/IGE ./.
Oh/NPP mond/IGE e1s/IK nekem/NM , hogy/KOT ki/IK vagyok/IGE ./, hogy/KOT azt/NM ko2vetsem/MN ./, mint/KOT vezeto3je1t/IGE a/DET vak/MN ./.
-/- Nem/HM tudom/IGE ./, hogy/KOT ki/IK vagy/KOT Suryakanta/MN kira1lyy/FN ./.
-/- Hiszen/KOT miniszterem/IGE vagy/KOT e1s/IK ifju1sa1gnak/FN talrsa/MN !/!
-/- Ifju1sa1gonak/IGE talrsa/MN vagyok/IGE Suryakanta/IGE ./: fe1rfiba1r1t/FN ./.
Tudom/IGE szavaidat/IGE e1s/IK tetteidt/FN tudom/IGE ./.
Te1ged/NM nem/HA tudlak/IGE ./.
-/- Nem/HA tudom/IGE ./, hogy/KOT ki/IK vagy/KOT Suryakanta/MN kira1lyy/FN ./.
-/- Hiszen/KOT miniszterem/IGE vagy/KOT e1s/IK ifju1sa1gnak/FN talrsa/MN !/!
-/- Ifju1sa1gonak/IGE talrsa/MN vagyok/IGE Suryakanta/IGE ./: fe1rfiba1r1t/FN ./.
Tudom/IGE szavaidat/IGE e1s/IK tetteidt/FN tudom/IGE ./.
Te1ged/NM nem/HA tudlak/IGE ./.
-/- Nie1r1t/HM szeret/IKE hali1t/HM engem/NM ./, jai/FN !/!
-/- Szavake1rt/IGE e1s/KOT tetteke1rt/FN szeret/IKE a/DET barai1t/FN ./, Suryakanta/HM kirai1lyy/FN ./.
Savak/FN e1s/KOT tettek/IGE mo2ge1/NU asszony/FN tud/IKE szeretni/INF ./.
-/- O1h/IGE bu2szke/MN Balapandita/IGE ./, hu3se1g1n/FN szekre1nye/FN !/!
-/- Ilgy/KOT kia1ltott/IGE fel/HA akkor/HM Suryakanta/IGE ./.
-/- Arck1lem/IGE elo3tt/NU u2lssz/IGE e1s/KOT valr1sz/IGE ra1m/NM e1s/KOT szemeden/FN o3rzo2d1FDN re1gi1MN arcomat/FN ./.
Szemedbo3l/FN fogom/IGE azt/NM elo3venni/INF ./, hogy/KOT u1jra/HM magamra/FN o2tisem/IGE ./.
Oh/NPP gyveru2nk/FN Razakosa/IGE ./, siessu2nk/FN Balapandita1hoz/IGE ./.
Akkor/NM Suryakanta/IGE kirai1lyy/FN elindult/IGE hazafele1/MN az/DET erdog3n/FN ./, o3t/NM pedig/KOT ko2vette/IKE elo3so3z/NM minisztere/FN e1s/KOT ifju1sa1gnak/FN talrsa/MN ./.
De/KOT ha/KOT Razakosa/FN nem/HM tudta/IKE volna/IKE ./, hogy/KOT ki/IK le1pked/IKE elo3tete1NU az/DET erde1/MN o2sve1nyn/FN ./, ja1r1sa1ro1l11IGE ra1m/NM nem/HM ismert/IKE volna/IKE Suryakanta/HM FNI1lyra/IGE ./.
Mert/KOT aki/NM me1g/HM tegnap/HA u1gy/HM ja1r1t/IKE mint/KOT a/DET vad/MN elefa1t/FN a/DET tang01/MN bozo1ltban/FN ./, az/DET ma1l1na1MN u1gy/HM ja1r1t/IKE mint/KOT az/DET u1jon/FN vedlett/IKE ki1gyo11IGE ./, amely/NM bo3re1t/FN fe1ltyve1n/FN kanyaroglva1FN keru2l1IGE minden/NM kavicsot/FN ./.
E1s/KOT aki/NM te1n/FN u1gy/HM haladt/IKE cell1ja/FN fele1/NU ./, mint/KOT Krishn/FN ki103tt/FN NY1la/IKE ./, az/DET tel1oval1za1VA/HM meg-mega1llt/IKE most/HM ./, jobbra-balra/HM ne1zve/HM ./, mint/KOT a/DET sera1lmes/FN asszony/FN aki/NM elo3so2r2r/HA szo2kik/FN kedve1hez/IKE e1s/KOT la1bperceinek/FN cso2rge1se/FN riastja1IGE az/DET e1j1/MN csendben/FN ./.
-/- O1h/IGE Suryakanta/IGE ./, Suryakanta/IGE mi/NM lett/IGE belo3led/NU !/!
Ma1r/HM esteledett/IKE mikor/HM a/DET palota/FN ele1/NU e1r1tke1IKE ./.
Ma1r/HM a/DET nap/FN ura1FN va1nlor1t1ja1nak/FN ve1ge1n/FN bete1r1t/IKE az/DET alkony/FN bi1bor/FN palota1ja1ba1FN de/KOT onnan/HM is/KOT ki1z3etve1n/IKE a/DET tengerbe/FN vetette/IKE magai1t/NM ./.
Ma1r/HM a/DET hold/FN ezu2stvize/FN csurgott/IKE le1IK a/DET palota/FN malachit/FN le1pcso31n/FN e1s/KOT a/DET tancolo1/MN palova1k/FN kapkodta1IKG e1lukat/FN ./, hogy/KOT meg/IK ne/HA a1zze1k/IKE ./.
Ma1r/HM minden/NM fa1klya/FN kialudt/IKE csak/HM a/DET legbelso3/MN szoba1ban/FN e1gett/IKE egy/DET illatoz1/MN lam1mpa/FN ./, mint/KOT a/DET so2te1t/FN palotna1KNM vi1lai1g1ito1FN sz11ve/FN ./.
Balapandita/FN virrasztott/IKE Suryakanta/IGE kira1lyy/FN arck1pe1FN elo3tt/NU ./.
Mikor/HM a/DET kira1lyy/FN remego3/MN ke1zzel/FN fe1r1hu1zta/IKE a/DET fu2go2ny3t/FN e1s/KOT a/DET szoba1ba1FN le1pett/IKE volna/IKE ./, a/DET bu2szke/MN Balapandita/FN felugrott/IKE az/DET oroszla1n1la1bu1/MN pa1l1ma1ro11FBN ./.
-/- Ki/IK vagy/KOT te1/NM ./, e1s/KOT hogy/KOT mersz/IKE a/DET kira1lynyo3/MN szoba1ja1ba1FN le1pni1INF ?/!
-/- Kia1ltotta/IGE hali1traszegett/IGE fejje1/FN haragosan/MN ./.
-/- Suryakanta/IGE vagyok/IKE ./, a/DET fe1r1jed/FN ./

125
Nem/HA ismersz/IGE ?!

-A/- Szemtelen/MN csalog1/FE
Suryakanta/FN hangja/FN olyan/NM mint/KOT a/DET bagzo1/FE tigrise1/FE ./.
De/KOT a/DET te/NM hangod/IGE mint/KOT a/DET here1lt/FE szogalke1/FE u1gy/HA remeg/FE ./.
Akkor/NM a/DET kiral1ly/FN egy/DET le1pelssel/FE ko2zelebbing1/FE le1pett/IGE a/DET la1mpalhox/FE e1s/KOT ilgy/KOT szo1lt/IGE :/

-/- Oh/NNP sze1csis1po3ju3/INF Balapandita/FN ismerj/IGE meg/IK ,/ , mert/KOT e1n/NM vagyok/IGE a/DET fe1ljed/FE Suryakanta/FN kiral1ly/IGE ./.
Akkor/NM Balapandita/FN egy/DET le1pelst/FE haltrale1pett/IGE e1s/KOT ilgy/KOT szo1lt/IGE :/ Suryakanta/INF arca/FN mint/KOT a/DET a/DET te/NM arcod/FE puha/NM e1s/KOT valito1oz1/INF mint/KOT a/DET sz1ne1sz1zek1/FE ./.

Jai/ISZ !/

-/- Oh/NNP Balapandita/FN ./, Balapandita/FN ismerj/IGE meg/IK ./!
E1n/NM vagyok/IGE Suryakanta/FN ./.

-/- Suryakanta/FN kiral1ly/FN ma/HA elindult/IGE az/DET erdo3be/FE vezekek1ini/INF e1s/KOT Suryakanta/FN szalnde1ka1t/FE szoha/HA meg/IK nem/HA ma1si1tja/IGE ./.
Jai/ISZ !/

-/- O1h/IGE Balapandita/FN visszate1ttem/IGE hozzal1d/INF ./, mert/KOT valgyo1d1tam/IGE utalnad/NM ./.

-/- Suryakanta/FN a/DET e1s/KOT ilgy/KOT e1s1ko3/NM e1s1ko3/NM ./, a/DET mozdulatlan/INF e1s/KOT ve1gso3/NM ./.
Nekem/FE kiel/IGE o3hozza1/IGE mennek/IGE ./.
O3/NM nem/HA jo2n/INF e1nhozza1/INF szoha/HA ./.

Jai/ISZ !/

-/- Eljo2ttem/IGE hozzal1d/INF ./, hogy/KOT talmasztd/INF meg/IK gyo2kereszakad/HA ingo1/IGE lelkekem/FE ./.
-/- Jai/ISZ Suryakanta/FN a/DET szikla/FE ./, melyhez/IGE az/DET e1n/NM lel kem/IGE ingo1/INF hajo1ja1t/FE ko2tto2ttem/IGE ./.
Akkor/NM a/DET kiral1ly/FN me1g/HA e1s/KOT egy/DET le1pelssel1/INF ko2zelebbing1/FE le1pett/IGE az/DET illatozo1/INF la1mpalhox/INF e1s/KOT ilgy/KOT szo1lt/IGE ./-

-/- O1h/IGE Balapandita/FN nelzz1/IGE ra1m/INF ./, nelzzed/IGE a/DET kiral1ly/FN e1lkszerek1/INF mellemen/INF ./.

E1n/NM vagyok/IGE a/DET fe1ljed/FE Suryakanta/FN ./.
Akkor/NM Balapandita/FN me1g/HA e1s/KOT le1pelssel/INF haltrale1pett/IGE e1s/KOT ilgy/KOT szo1lt/IGE ./, a/DET be1n/NM ne1zz1/IGE ra1m/INF e1lkszerek1/INF mellemen/INF ./.

E1n/NM vagyok/IGE a/DET fe1ljed/FE Suryakanta/FN ./.
Akkor/NM Balapandita/FN me1g/HA e1s/KOT le1pelssel/INF haltrale1pett/IGE e1s/KOT ilgy/KOT szo1lt/IGE ./ -/- Ha/KOT te/NM Suryakanta/FN neve1lt/INF e1s/KOT kiral1ly1/INF MN e1szene1ft/INF veiseled/IGE akkort/HA Suryakanta/FN MN helye1re/INF le1pel11/INF ebben/INF az/DET e1le1tben/INF e1s/KOT Suryakanta/FN kiral1ly/FN ./, a/DET fe1rrfitigris/INF ./, az/e1n/NM fe1rjemen/INF meghalt/IGE ./.

Ezt/INF mondta/IGE Balapandita/FN e1s/KOT bu2szke1/INF arca1t/INF kezeke1/INF VK lo1tusza1/INF mo2ge1/INF rejt1/INF lelkem/INF bizodalma/IGE ./!

-/- O1h/IGE Balapandita/FN jai/IGE nekem/INF ./!

Te1gedz/INF a/DET hu3se1/INF szekre1nye1nek/INF hivnak/IGE e1s/KOT szerelmed/INF a/DET ko1llando1s1galban/INF volt/INF ingo1/HA lel kem/INF bizodalma/IGE ./!
Akkor/NM Balapandita/FN bu2szke1/INF kiegyenesedett/IGE ./, -/- Bizony/HA holtomig/INF hu3/MN maradok/MN Suryakanta1hoz/INF kinek/INF sz1m1ve1tet/INF adtam/IGE ./.

Suryakanta1hoz/INF ko1ll/hu3se1/INF szekre1nye1nek/INF hivnak/IGE e1s/KOT szerelmed/INF a/DET ko1llando1s1galban/INF volt/INF ingo1/HA lel kem/INF bizodalma/IGE ./!

Akkor/NM Balapandita/FN bu2szke1/INF kiegyenesedett/IGE ./, -/- Bizony/HA holtomig/INF hu3/MN maradok/MN Suryakanta1hoz/INF kinek/INF sz1m1ve1tet/INF adtam/IGE ./.

Jai/ISZ !/

-/- Balapandita/FN jai/IGE nekem/INF ./!

Suryakanta1hoz/INF ko1ll/hu3se1/INF szekre1nye1nak/INF hivnak/IGE e1s/KOT szerelmed/INF a/DET ko1llando1s1galban/INF volt/INF ingo1/HA lel kem/INF bizodalma/IGE ./!

Akkor/NM Balapandita/FN bu2szke1/INF kiegyenesedett/IGE ./, -/- Bizony/HA holtomig/INF hu3/MN maradok/MN Suryakanta1hoz/INF kinek/INF sz1m1ve1tet/INF adtam/IGE ./.

Jai/ISZ !/
hangja/FN mint/KOT a/DET here1lt/FN szolga1kel/IGE ./, viselkedel1se/FN mint/KOT a/DET terhes/FN asszonyokel1/FN ./, olyai/NM ./.
Ezt/NM mondva1n/FN felrehu1alta/IGE az/DET ajto1/FN arannya/IGE hi1mzett/FN fu2ggo2nyel1/IGE e1s/KOT kikiallott/IGE iy/: 
-/- Oh/NNP Cseti/FN ./, kedves/MN csele1dem/MN e1lbredji/FN ./.
Fehe1r/MN ruha1lt/FN hozz/FN nekem/NM e1s/KOT szeza1mmagokat/FN kel1szi1ts/FN v1lzzel/FN e1s/KOT puha/MN darbha-fu2vet/FN ./.
Akkor/NM Cseti/FN megjelent/IGE az/DET aranyfu2ggo2ny/FN nyi1la1sa1ban/FN e1s/KOT allomto1l/FN talgult/MN szemekkel/FN ne1lzett/IGE a/DET szoba1ba/FN e1s/KOT halkan/MN szol1lt/IGE mint/KOT aki/NM fe1ll/SZN ./, hogy/KOT ve1ll/Iy e1llma1bo1ll/FN o2mmagalt/NM fe1lbresz1tt/NM iy/: 
-/- Oh/FN bu2szke/MN Balkapandita/IGE a/DET fehe1r/MN ruha/FN o2zvegyek/FN ruha1ja/FN e1s/KOT szeza1mmagok/FN puha/MN darbha/IGE fu3vel/FN halotti/MN alldozat/FN ./.
Mie1rt/HA ki1valn/IGE mindezt/NM o1h/MN kirallyno3m/MN mikor/FN ott/HA la1tom/IGE a/DET fe1rfit/FN kinek/NM melle1n/FN a/DET kiralliyi/FN el1kszerek/MN csillognak/FN ?/?.
-/- ADET kiralliyi/MN e1kszerek/FN hordja/IGE me1g/HA a/DET test/FN de/KOT a/DET lelket/FN belo3le/NU kite1pte/IGE Yama/FN a/DET halall/FN ura/FN ko2te1llhuurokkal/FN maga/NM uta1n/NU hu1lzva1n/IGE ./.
Szo1l1tsd/FN szolg1limat/FN kik/FN atya1m/FN halza1bo1l/FN jo2ttek/Iy velem/NM iy./, hogy/KOT engem/NM hozza1/HA visszavignyek/Iy ./.
Atya1m/FN kel1szi1tsen/MN ma1glyva1t/MN sa1lmonra/FN ./, hogy/KOT a/DET tu3z/FN fu2stje1vel/FN sa1lhhassak/Iy a/DET meghalt/Iy Suryakanta/Iy uta1n/NU ./, akihez/NM o2ro2kke1l/HA hu3/MN maradok/FN ./.

127
Texts with PoS and subtags (tags for inflectional properties)

Ko2zel/MN hulzoidik/IGE_e3 ellemtemez/FN_PSe3i_ACC ALL /, s/KOT nem/HA tudom/IGE_Tel1 /, mekkora/FN_SUB /, hol/HA velgzo3dik/IGE_e3 /, fal/FN nekem/NGM_DAT o3/NM /, e1l03/MN panaszfalom/FN /, A/DET ritku1o1/MN levelesen/MN_ESS_MOD althullo1/MN fe1inyt/FN_ACC elke1pzelem/FN /, a/DET to2lgy/FN /, bu2kk/IGE_e3 /, szíflalevelek/FN_PL fodroztza/IGE_TMe3 fe1inyt/FN_ACC /, az/DET e3o3utan1i/INF poosolya1ko1h0z/FN_ALL hasonlo1/MN avant/IGE_Me3 /, vadleso3/MN magasalaltait/FN_PSe3i_ACC /, amelyek/NM_PL tekinttem/IGE_TMe1 fo2lvetezned/1k/IGE_Fe1 az/DET e1g/FN fa1tylaira/FN_PSe3i_SUB /.

Izgat/FN_ACC a/DET megoldott/MN rejet1lyu3/MN szalton1k/FN_PL uta1n/NM /, szeretne1k/IGE_Fe1 odafutni/INF a/DET to2bbiekto3l/FN_ABL /, szeretne1k/IGE_Fe1 meglemsni/INF egy/FN vadat/FN /, hajtami/INF /, elveszteni/INF e1s/KOT u1tra/FN_HA megtalaltam/INF /.

Szeretne1k/IGE_Fe1 elveszni/INF e1s/KOT si1rni/INF /, elo3keru2lni/INF /, ra1hajolni/INF a/DET koraregelel/MN tu3zhely/FN melege1re/FN_PSe3_SUB a/DET hazate1r/MN hadifogyok/FN_PL o2ro2me1vel/FN_PSe3_INS /.

Vonz/MN /, mert/KOT titkait/FN_PSe3iACC sohasem/HA fedte/IGE_TMe3 elo3ttem/IGE_TMe1 e1s/KOT sohasem/HA hagytak/IGE_Mt3 /, hogy/KOT titkait/FN_PSe3i_ACC megta1lljam/FN /

A/DET bozo1tbo1l/FN_ELA kiugro1/MN szarvast/FN_ACC ki/IK la1tta/I1G_TMe3 ?./

Igaz/MN /, olyan/NM mint/KOT a/DET szappanrekla1mon/FN_SUP /, elo3/SZN la1bait/FN_PSe3i_ACC maga/NM ala1/NU hulzv2a/HA /, szarvait/FN_PSe3i_ACC e1gre/FN_SUP vetve/HA szok2ell/FN /, repu2/I2G olyan/NM fo2nse1ges/MN_ESS_MOD /, hogy/KOT lelo3ve/HA is/KOT u1gy/HA bukik/IGE_e3 /, ahogya/HA a/DET szabadal3/INF nagy/FN lejala1k/FN_PSe3_DAT bukni/INF illik/FN /.

La1ttam/IGE_TMe1 a/DET szarvast/FN_ACC rabnak/MN_FOK_DAT /, fiatal/MN e1s/KOT megte1pafalzott/MN volt/I1G_Me3 /, mint/KOT akut/NM ACC e1lve/HA fognak/I1G_Te1 el/IK /, mint/KOT akut/NM ACC ero3szakos/MN kezkek/FN_PL_INS /, te1rdekkel/FN_PL_INS magukel1va1/NM_PSt3iPOS_FAC tesznek/I1G_Te3 /.

Nem/HA la1ttam/IGE_TMe1 szabadnak/I1G_Te3 /.

Ha/KOT nem/HA lettem/I1G_TMe1 olyan/NM szabandonfuto1/MN /, szabodon/MN_SUP do3zo2l03/MN /, szabadon/MN_SUP szenvedo3/MN /, hait/Ha megatlani/INF ak1hisztal/FN_TMe1 legala1bb/Ha /.

Mert/KOT csak/HA a/DET gyalva/MN nyulat/FN_ACC /, a/DET sompolgyo1/MN ro1kakat1/FN_ACC /, a/DET szetszalado1/MN egereket/FN_PL_ACC ismerem/I1G_Te1 /.

Puska1val/FN_PSe3_INS menjek/FN_PL oda/I1K ?./

Mal/I1R HA csak/HA fegyverrel/FN_INS juthatok/FN_PL oda/I1K /, ahov/a/HA valamikor/HA u2res/MN ke1zze1/FN_INS akartam/I1G_TMe1 ?./

Ilgy/KOT nyeriek/FN_PL biztonsalgot/FN_ACC /, lo2veszen/INF /, leseko3dtessen/I1G_MN_ESS_MOD /.

Hulzoon/FN_SUP /, vezessen/INF /, leseko3dtessen/I1G_MN_ESS_MOD /, lo2veszen/I1G_MN_ESS_MOD /.

Borra1/FN_INS e1s/KOT friss/MN vadhu1ssel/FN_INS megki1naltozson/I1G_Pe3 ?./

Engedjen/I1G_Pe3 bo3ge1st/FN_ACC hallani/INF /, la1tni/INF hajnali/INF forra1/FN_ACC /, megtudni/INF titkot/FN_ACC /.

Nem/HA kell/I1G nekem/NGM_DAT a/DET kutatalso1/FN_ELAL passzio1/FN /, vesze1lybo3l/FN_PSe3_ELAL se1talga1tatlal/FN /.

Vagy/KOT nincs/I1G ma1r/Ha ez/NM az/DET erdo3/FN /.

Vagy/KOT talal1n/Ha benne/Ha vagyok/I1G_e1 ?./

S/KOT mindig/HA egyma1rsa/NM_SUB gondolunk/I1G_Te1 /, amint/Ha ja1rom/I1G_Te1 o2svelnyeit/FN_PSe3i_ACC egyedul2/HA /, u2res/MN ke1zzel/FN_INS /.
Lendu2l/MN_ESS_MOD a/DET hab/FN s/KOT a/DET part/FN fala1n/FN_PSe3_SUP zu2stsarkanyu1s/MN la1ba/FN_PSe3 dobban/MN_FOK_ESS_MOD /
ne1zd/IGE_TPe2 ,/ fu2rdik/IGE_e3 a/DET fekete/MN la1ny/FN ,/ fekete/MN la1ny/FN fehe1r/MN habokban/FN_PL_INE ./.
Elszenderu2lt/IGE_Me3 a/DET bu1/MN szeme1n/FN_PSe3_SUP ,/ hulla1mba/FN_ILL hull/FN ma/HA teste/FN_PSe3 ,/ lelke/FN_PSe3 ,/ hulla1mos/MN haj1u1/MN vo3lege1ny/FN milyen/NM ero3sen/MN_ESS_MOD a1to2lelte/IGE_TMe3 ./
De/KOT ne1zd/IGE_TPe2 :/ so2te1t/MN erdo3k/FN_PL ko2zo2tt/NU nagy/MN ,/ ordas/MN fellegek/FN_PL szakadnak/IGE_t3 s/KOT jo2n/IGE alra/FN_PSe3 zalporo2tt/MN go2rgeteges/MN hegyi/MN pataknak/FN_DAT ./
Haragos/MN ,/ szennyes/MN a1radat/FN ,/ a/DET medre1t/IGE_Pe3_ACC karmok/FN_PL a1ssa1k/IGE_TP_t3 ./
Ke1rdu2lt/IGE_TPe2 meg/IK a/DET ho2kent/FN_CAU gal1takat/FN_PL_ACC ,/ gal1ncsoljakk/IGE_Tt3 a/DET vizek/IGE_e1 futas1sa1t/FN_PSe3_ACC ./
Ne/HA essen/MN_ESS_MOD folt/FN fe1nyes/MN haja1n/FN_PSe3 SUP ,/ izsapos/MN a1r/FN hozzal/DEMN_ALL ne/HA e1rjen/IGE_Pe3 :/ fekete/MN la1ny/FN talan1n/HUa ,/ talan1n/HU utolszor/SZN fu2rdik/IGE_e3 /- ho1fehe1ren/MN_ESS_MOD ./

BALA1ZS/FN BE1LA/FN :/ A/DET HA1ROM/FN_HU3SE1GES/FN KIRA1LYLE1A/NY/FN
El1t/IGE_Me3 egyszet/HA Kandrapura/FN_SUB val1rosalban/FN_INE egy/DET Suryakantana/FN nevu3/MN hatalmas/MN kir1ly/FN ,/ aki/NM ha1rom/SZN szex tulajdonsa1ga/IGE_TPe2 a1lta1/NU olyan/NM volt/IGE_Me3 a/DET fe1rifikak/FN_PL ko2zo2tt/NU mint/KOT a/DET ko2vek/FN_PL ko2zt/NU az/DEMN Adamanth/NNP :/ tu2ndo2kko3/MN ./, tiszta/MN e1s/KOT ke1melny/FN ./
Tu2ndo2kko3/MN sze1pse1ge/FN_PSe3 minden/NM asszonyt/FN_SUP harmon/MN mint/KOT a/DET ko2rzak/FN_SUB ./
Bu3ntelen/MN tisztasa1ga/FN_PSe3 emberfeletti/MN ero3t/FN_ACC adott/IGE_Me3 neki/NM_DAT ./
De/KOT keme1nyse1ge/FN_PSe3 volt/IGE_Me3 a/DET legnagyobb/MN_FOK ./
Lelek1nek/FN_PSe3_DAT falja/FN_PSe3 elo3bbi/MN e1letekben/FN_PL_INE gyo2kereketz/IGE_Me3 e1s/KOT eze1r1/KOT mozdi1ghtatatlan/MN vir1gokke1nt/NF bottomek/IGE_Mt3 rajta/NM_SUP tettek/IGE_Mt3 e1s/KOT szavak/FN_PL ./, melvek/FN_PL ezer/SZN e1/v/FN o1ta/NM ke1szen/MN_ESS_MOD voltak/IGE_Mt3 ./
Suryakantana/FN_TMe3 kir1ly/FN csak/HA hullatta/IGE_TMe3 o3ket/NM_ACC ,/, de/KOT rajtuk/NM_SUP nem/HA val1loztatatott/IGE_Me3 ./
Suryakantana/FN_TMe3 kir1ly/FN uta1lta/I G E_M t3 a/DET szavakat/FN_PL_ACC e1s/KOT minden/NM ke1rde1sre/FN_SUB tettek/IGE_Mt3 felelt/IGE_Me3 ./
A/DET ko2nyo2rgo3k/FN_PL el1s/KOT ke1relmez03k/FN_PL felett/NU ne1ma1n/FN_PSe3_SUP ne1zett/IGE_Me3 el/IK mozdulatan/MN arccal/FN_INE de/KOT mire/HA hazate1rtek/IGE_Mt3 ke1rel1su2k/FN_PSt3 teljes1ltve/HA volt/IGE_Me3 ./
Suryakantana/FN_TMe3 kir1ly/FN u1gy/HA a1ltt/IGE_Me3 az/DEMN emberek/FN_PL ko2zo2tt/NU mint/KOT az/DEMN erdo3/FN te1tova/MN hajlongo1/MN no2ve1nyei/FN_PSe3i ko2zo2tt/NU a/DET merev/MN istenszobor/FN ./, melyben/FN_PSe3_INE a/DET szellef/IGE_TPe2 e1r1bho2rto2nebe/FN_ILL za1rva/HA o2ro2k/MN ido3kre/FN_SUB egyforma/MN e1s/KOT mozdultatlan/MN ./
Egyszet/HA Suryakantana/FN_TMe3 kir1ly/FN maga1hoz/NM_ALL hivatta/IGE_TMe3 feles1ge11/FN_PSe3_ACC a/DET bu2szke/MN Balapandita1/FN_ACC ./
Mikor/HA Balapandita/IGE_TMe3 a/DET terembe/FN_ILL le1pett/IGE_Me3 azon/NM halk/MN remeges1/FN futott/IGE_Me3 vel1gik/IK mint/KOT a/DET fu2ves/MN re1ten/FN_SUP ha/KOT a/DET felkelo3/MN hold/MN re1lelhe1/FN_INS ./
Mert/KOT Balapandita/IGE_TMe3 szel1pse1ge1ben/FN_PSe3_INE a/DET hold/MN go2mo2lyu3se1ge/FN_PSe3 ./, futol1inda/FN hajla1sa/FN_PSe3 ./, fu3/FN
remege1se/FN_PSe3 ./, leve1he/FN_PL ra1jazsa1sa/FN_PSe3 e1s/KOT a/DET papagal1/IrG_Pe2 melle/FN_PSe3 la1gy1sa1ga/FN_PSe3 egyesu2lek/IrG_Mt3 mint/KOT egy1ma1t/NM_ACC re1g/MN keresu3/MN szerelemek/FN_PL ./, -/ Mie1r/HA hi1vat/FN_ACC uralm/FN_PSe3 ./, Suryakanta/IrG_TMe3 kira1ly/FN ./, -/ ke1rdezt/IrG_TMe3 Balapandita/IrG_TMe3 ./.
De/KOT Suryakanta/IrG_TMe3 nem/HA felel1/IrG_Me3 hanem/KOT odale1ptet/IrG_Me3 a/DET sze1pcs1po3ju2ho2zz/FN_ALL e1s/KOT homlokon/FN_PL_SUP cso1kolsz/IrG_e2 e1s/KOT olyan/NM jeleket/FN_PL_SEC teszel/IrG_e2 mint/KOT a/DET bulcsu1z/FN_PL ./.
-/- Uram/FN Suryakanta/IrG_TMe3 ./- szo11t/IrG_Me3 Balapandita/IrG_TMe3 akkor/HA ./- homlokon/FN_PL_SUP cso1kolsz/IrG_e2 e1s/KOT olyan/NM jeleket/FN_PL_SEC teszel/IrG_e2 mint/KOT a/DET bulcsu1z/FN_PL ./.
Vajon/KOT bulcsu1z/FN_PLINF akarsz/IrG_e2 a/DET lo1kemekte/NM_ALL te/NM fe1rfitigris/FN ?/?
Suryakanta/IrG_TMe3 nem/HA felel1/IrG_Me3 hanem/KOT elso3/SZN minisztere1hez/FN_PSe3_All e1s/KOT ifjusai1nak/FN_PSe3_DAT tarsu2lho2zz/FN_ALL fordult/IrG_Me3 ./, mondva1n/FN_PSe3_SUP ./- /-
Razakosa/FN_PSe3 ke1szu2lj/IrG_Pe2 u1tra/FN_SUB ./.
De/KOT ki1s1rel1/IrG_TM1 ne/HA ko2vessen/MN_ESS_MOD e1s/KOT fullajta1rok/FN_PL elo1tu2nka/IrG_T1 ne/HA fussenak/IrG_T3 ./, E/NM szavako1b/FN_PL_ELA Balapandita/IrG_TMe3 e1s/KOT Razakosa/IrG_PSe3 mege1rel1te1kk/IrG_TMe3 hogy/KOT Suryakanta/IrG_TMe3 kira1ly/FN az/DET erdo3be/FN_PL nad1/IrG_Val valamely/FN szent/MN liget/FN_ACC la1togata1sa1la/IrG_PSe3_SUB ./, hogy/KOT a1llozta1l/FN_INS ./, fu2drel1ssel/FN_INS e1s/KOT medita1cio1val/FN_PSe3_INS tiszt1tsa/IrG_TPe3 lelke1t/FN_PSe3_ACC e1s/KOT ereje1t/IrG_TPe3 e1s/KOT neve1z/IrG_TPe3 ACC no2velje/IrG_TPe3 ./.
Akkor/NM_TEM Balapandita/IrG_TMe3 hangja1ra/FN_PSe3_SUB nedves/MN fa1yolt/IrG_Me3 borit1/IrG_TMe3 e1s/KOT ify1gyo/HA szo11t/IrG_Me3 ./- /-
Bizony/HA vissza1jot/IrG_PSe3 u1tra/FN_SUB ./.
Megfe1setem/IrG_TMe3 ke1lpedet/FN_ACC ./, hogy/KOT a/DET hold/MN vilai1g1ts1son/FN_SUP lekemere/FN_SUB mi1g/HA napja/FN_PSe3 vi1szaze1r/FN ./.
De/KOT Suryakanta/IrG_TMe3 kira1ly/FN nem/HA felel1/IrG_Me3 hanem/KOT ne1ma/MN ajakkal/FN_PL_INS e1s/KOT besze1lo3/MN(szemmel/FN_INS ne1zte/IrG_TMe3 fe1sere1gt/IrG_PSe3_ACC akit/NM_ACC a/DET hu3s1eg/FN szkre1nye1nek/FN_PSe3_DAT nevezte/IrG_Mt3 azta1n/HA elindult/IrG_Me3 a/DET palota/FN_Kapuja/IrG_Te3 bele1/NU ./.
O3t/NM_ACC pedig/KOT ko2vet1/IrG_TMe3 Razakosa/IrG_PSe3 elseo3/SZN minisztere1re/FN_PSe3 e1s/KOT ifjusai1nak/FN_PSe3_DAT ta1rsa/FN_PSe3 ./.
Mikor/IrG_Me3 ma1r/HA mel1yen/MN_ESS_MOD ja1rta/IrG_Mt3 az/DET erdo3ben/FN_INE ahol/HA a/DET_Kadamba/IrG_TMe3 ILL fa/FN_iflata1to1l/IrG_PSe3_ABL megdu2ho2do2zt/MN pe1zsmaszarvasok/IrG_PL tu1rja1k/IrG_Tt3 magukatu/NM_PSe3 ACC a/DET szirimhullato1/MN ja1szimnobrokony/FN_PL_SUP keresztu2l/HA ./, Suryakanta/IrG_TMe3 kira1ly/FN hi1telen/MN mega1l1t/IrG_Me3 ./, -/ Mit/NM_ACC la1tta1/IrG_Me2 uralm/FN_PSe3 ./, -/- ke1rddezte/IrG_TMe3 Razakosa/FN_PSe3 ./.
Suryakanta/IrG_TMe3 kira1ly/FN akkor/HA lehajolt/IrG_Me3 e1s/KOT mikor/IrG_Me3 megint/HN kiegyenesedette/IrG_Me3 jobb/MN_POk keze1vel/FN_PSe3_INS egye/DET nagy/MN fekete/MN_ki1gyo1/MN_torka1/IrG_PSe3_ACC fo1ta/IrG_TMe3 ./, bal/MN keze1vel/FN_PSe3_INS pedig/KOT egye/DET kis/MN nyulat/FN_ACC ./, melyet/FN_PSe3 a/DET ki1gyo1/MN_sza1ja1bo1/IrG_PSe3_ELAV ett/IrG_Me3 ki/IK ./.
Azuta1n/HA mind/NM a/DET ketto3t/SZN_ACC eldobta/IrG_TMe3 e1s/KOT toval1b/IK akart/IrG_Me3 menny/NM/AccINF ./.
Hanem/FN akkor/HA a/DET fekete/MN ki1gyo1/FN felajugodott/IGE_Me3 elo3tte/NM_Pl rendelte1k/IGE_TMt3 talplale1konomi/MN_ESS_MOD e1s/KOT ez/NM a/DET falaj/IGE_ACC melyet/IGE_AccountantN1 nem/HA szalajambo1l/IGE_PSE3_ELA kivette1l/IGE_Me2 e1hala1lom/IGE_Te1 elo3l/NU volt/IGE_Me3 utolsol1/MN monokve1l/IGE_AccFQ ./.

Mert/KOT tud/IGE_TPE2 meg/IK o1h/MN Suryakanta/IGE_TMMe3 kirally/FN ./, hogy/KOT harmics/SZN napig/FG_TER nem/HA ettem/IGE_TMMe1 e1s/KOT a/DET halail/IGE_Nfonk farka/IGE_Nfonk_1st ev/IGE_fejemen/FG_PSE3_SUB ./.

Tudsz/IGE_e2 e/NM nekem/NM_DAT ma1s/NM talplale1kot/IGE_Nfonk adni/INF a/DET nyu1l/IGE_Nfonk helyett/NM ./.

Suryakanta/IGE_TMMe3 kirally/FN mozdulatlan/MN arcfl/IGE_Nfonk INS ne1zte/IGE_TMMe3 a/DET fekete/MN ki1gyo1/FN ACC e1s/KOT nem/HA felelt/IGE_Me3 ./, hanem/KOT kihile1ta/IGE_TMMe3 to3re1l/NU_ABL e1s/KOT izmos/MN barna/MN karjalbo1l/IGE_PSE3_ELA kanyaritott/IGE_Me3 egy/DET darabot/IGE_Nfonk e1s/KOT azt/NM ACC odadobta/IGE_TMMe3 a/DET ki1gyo1/FN e1l/NU ./.

Hanem/FN a/DET ki1gyo1/MN nem/HA nyul1t/IGE_Me3 utalna/NM ./.

Suryakanta/IGE_TMMe3 kirally/FN mozdulatlan/MN arcfl/IGE_Nfonk INS ne1zte/IGE_TMMe3 a/DET hosszul1/MN ./, fekete/MN hullal1t/IGE_ACC ./.

Lelek1t/IGE_PSE3_ACC oly/NM leve1ssel1/MN_FAC illette/IGE_TMMe3 a/DET halail/IGE_Nfonk mint/KOT a/DET vad/IGE_Me3 ele1ativon/FG_ACC az/DET erdei/MN leve1l/IGE_Nfonk mely/IGE_Me3 to3re1l/NU_ABL levajt/IGE_Me3 kanyari1tott/IGE_Me3 ./.

Elpen/HN alt/IK akart/IGE_Me3 le1pmi/INF rajta/NM SUP ./, hagya/KOT tovalbbmenjen/IGE_Me3 mikor/HA hirtelen/MN megjelent/IGE_Me3 elo3tte/NM Ganesa/IGE_TO_Fonek az/DET ele1ativoncubu1/MN istic/IGE ./.

O1rals/IGE_PSE3i_SUP altizott/IGE_Me3 a/DET nap/INF ./, hagya/KOT tu2zes/IGE_MIN korallbokro1kkel1/MN_FORE pirosoltat/IGE_Me3 benne/HA az/DET erek/IGE_Nfonk_PL ./.

Rolzasszili/IGE_PSE3_SUP suga1r/FN omotma1nyalt/IGE_PSE3_Nfonk talnban/IGE_PSE3_NACC emelte/IGE_TMMe3 e1s/KOT ilgy/KOT szol1lt/IGE_Me3 ./.

Suryakanta/IGE_TMMe3 o2lte1l/IGE_Me3 ./.

Suryakanta/IGE_TMMe3 mozdulatlan/MN szemmel1/MN_INS ne1zett/IGE_Me3 Ganesa/IGE_Me3akis/MN szeme1be1/IGE_Nillon_1st e1s/KOT ilgy/KOT felelt/IGE_Me3 ./.

Az/DE3 isolte/NM_Pl o2lte1k/IGE_Me3 akik/NM_Pl termel1szetmet1/IGE_Nfonk e1gy/Hフォ Amer1ta1k/IGE_TMMe3 hogy/KOT utaljan/FG_Nfonk e1s/KOT szavakat/IGE_PSE3_INS e1s/KOT ketekkel/IGE_Nfonk_PSE3_INS beszekel1jek/IGE_PL ./.

Lekken/FG viralga/FG_PSE3 elo3bbi/MN e1lekteken/FG_PSE3_INET gyo2keredzik/IGE e1s/KOT sem/IGE_Me3 mozdu1n1/INF sem/IGE_Me3 valto1zni/INF nem/HA tud/IGE ./.

Akkor/NM_TEM Ganesa/IGE_Me3 nevetett/IGE_Me3 ./.

Az/DE3 iktek/IGE_PSE3 Succ/IGE_PSE3_Minemeg/IGE_PSE3_MIN algy/Hフォ Amer1t1/IGE_TMMe3 hogy/KOT utaljjan/FG_Nfonk e1s/KOT szavakat/IGE_PSE3_INS e1s/KOT ketekkel/IGE_Nfonk_PSE3_INS beszekel1jek/IGE_PL ./.

Avval/IGE_INS eltut3nt/IGE_Me3 ./.
A/DET kira1ly/FN melle1hez/FN_PSe3_ALL kapott/IKE_Me3 mintha/KOT nyi1l/FN fu1rta/IGE_TMe3 volna/IGE_Fe3 a1t/IK e1s/KOT felkiallott/IGE_Me3 :/:-
O1l/FN jat/IKE_PSe2 nekem/NM_DAT !/!

Akkor/NM_TEM Razakosa/FN_PSe3 kiejtette/IGE_TMe3 keze1bo3l/FN_PSe3_ELA i1jat/IGE_TMe3 keze1bo3l/FN_PSe3_ACC ./, mert/KOT soha/HA senki/FN me1g/HA Suryakanta/IGE_TMe3 kira1ly/FN jajsza1vatt/IGE_TMe3 sup lesu2to2tte/IGE_TMe3 ilyen/NM_ACC mert/KOT Suryakanta/IGE_TMe3 a1t/IK e1s/KOT felkiallott/IGE_Me3 ./.

O1h/FN jaj/IGE_Pe2 nekem/NM_DAT !/!

Akkor/NM_TEM Razakosa/FN_PSe3 kiejtette/IGE_TMe3 keze1bo3l/FN_PSe3_ELA i1jat/IGE_TMe3 keze1bo3l/FN_PSe3_ACC ./, mert/KOT soha/HA senki/FN me1g/HA Suryakanta/IGE_TMe3 kira1ly/FN jajsza1vatt/IGE_TMe3 sup lesu2to2tte/IGE_TMe3 ilyen/NM_ACC mert/KOT Suryakanta/IGE_TMe3 a1t/IK e1s/KOT felkiallott/IGE_Me3 ./.

Uram/FN Suryakanta/IGE_TMe3 kiejtette/IGE_TMe3 keze1bo3l/FN_PSe3_ELA i1jat/IGE_TMe3 keze1bo3l/FN_PSe3_ACC ./, mert/KOT soha/HA senki/FN me1g/HA Suryakanta/IGE_TMe3 kira1ly/FN jajsza1vatt/IGE_TMe3 sup lesu2to2tte/IGE_TMe3 ilyen/NM_ACC mert/KOT Suryakanta/IGE_TMe3 a1t/IK e1s/KOT felkiallott/IGE_Me3 ./.

O1h/FN jaj/IGE_Pe2 nekem/NM_DAT !/!

Akkor/NM_TEM Razakosa/FN_PSe3 kiejtette/IGE_TMe3 keze1bo3l/FN_PSe3_ELA i1jat/IGE_TMe3 keze1bo3l/FN_PSe3_ACC ./, mert/KOT soha/HA senki/FN me1g/HA Suryakanta/IGE_TMe3 kira1ly/FN jajsza1vatt/IGE_TMe3 sup lesu2to2tte/IGE_TMe3 ilyen/NM_ACC mert/KOT Suryakanta/IGE_TMe3 a1t/IK e1s/KOT felkiallott/IGE_Me3 ./.

Uram/FN /, mi/NM baj/FN e1rt/IGE_Me3 ?/?

Suryakanta/IGE_TMe3 ijedten/FN_INE fordult/IGE_TMe3 fele1/NU e1s/KOT me1lyen/MN_ESS_MOD elpirult/IGE_Me3 ./.

Uram/FN, /, mi/NM baj/FN e1rt/IGE_Me3 ?/?

Suryakanta/IGE_TMe3 ijedten/FN_INE fordult/IGE_TMe3 fele1/NU e1s/KOT me1lyen/MN_ESS_MOD elpirult/IGE_Me3 ./.

Uram/FN, /, mi/NM baj/FN e1rt/IGE_Me3 ?/?

Suryakanta/IGE_TMe3 ijedten/FN_INE fordult/IGE_TMe3 fele1/NU e1s/KOT me1lyen/MN_ESS_MOD elpirult/IGE_Me3 ./.

Uram/FN, /, mi/NM baj/FN e1rt/IGE_Me3 ?/?

Suryakanta/IGE_TMe3 ijedten/FN_INE fordult/IGE_TMe3 fele1/NU e1s/KOT me1lyen/MN_ESS_MOD elpirult/IGE_Me3 ./.

Uram/FN, /, mi/NM baj/FN e1rt/IGE_Me3 ?/?

Suryakanta/IGE_TMe3 ijedten/FN_INE fordult/IGE_TMe3 fele1/NU e1s/KOT me1lyen/MN_ESS_MOD elpirult/IGE_Me3 ./.

Uram/FN, /, mi/NM baj/FN e1rt/IGE_Me3 ?/?

Suryakanta/IGE_TMe3 ijedten/FN_INE fordult/IGE_TMe3 fele1/NU e1s/KOT me1lyen/MN_ESS_MOD elpirult/IGE_Me3 ./.

Uram/FN, /, mi/NM baj/FN e1rt/IGE_Me3 ?/?

Suryakanta/IGE_TMe3 ijedten/FN_INE fordult/IGE_TMe3 fele1/NU e1s/KOT me1lyen/MN_ESS_MOD elpirult/IGE_Me3 ./.

Uram/FN, /, mi/NM baj/FN e1rt/IGE_Me3 ?/?

Suryakanta/IGE_TMe3 ijedten/FN_INE fordult/IGE_TMe3 fele1/NU e1s/KOT me1lyen/MN_ESS_MOD elpirult/IGE_Me3 ./.

Uram/FN, /, mi/NM baj/FN e1rt/IGE_Me3 ?/?

Suryakanta/IGE_TMe3 ijedten/FN_INE fordult/IGE_TMe3 fele1/NU e1s/KOT me1lyen/MN_ESS_MOD elpirult/IGE_Me3 ./.

Uram/FN, /, mi/NM baj/FN e1rt/IGE_Me3 ?/?
Kedves Razakosa


A képletének alapja lenne az érthető és írható képletének alapja lenne.
Szemedbozó/FN_ELA fogom/IGE_Te1 azt/NM_ACC elo3venni/INF , hogy/KOT u1jrja/HA magamra/FN_SUB o2ltsem/INF ./.
Oh/NNP gy eru2nk/IGE_t1 Razakosa/FN_PSe3 ./, sieszu2nk/IGE_t1
Balapandita1hoz/FN_FON ./.
Akkor/NM_TEM Suryakanta/IGE_TMe3 kira1ly/FN elindult/IGE_Me3 hazafele1/HA az/DET erdo3n/FN_SUP ,/, o3t/NM_ACC pedig/KOT ko2vette/IGE_TMe3 elo3/SZN minisztere/FN_PSe3 e1s/KOT ifju1sal1gnak/FN_PSe3_DAT ta1r1sa/FN_PSe3 ./.
De/KOT ha/KOT Razakosa/FN_PSe3 nem/HA tudta/IGE_TMe3 volna/FN_Fe3 ./, hogy/KOT ki/IK le1pked/IGE elo3tte/NU az/DET erde1/MN o2sv1nyen/FN_SUP ,/, ja1r1sal1r1/FN_PSe3_DEL ral1/HA nem/HA ismert/IGE_Me3 volna/FN_Fe3 Suryakanta/IGE_TMe3_kira1lyra/FN_ALL ./.
Mert/KOT aki/NM me1g/HA tegnap/HA jai1rta/FN_IGE_Me3 mint/KOT a/DET vad/MN elefa1n1t/IGE_Me3 a/DET tango1/MN bozo1tban/FN_FON ,/, az/DET malma1/FN_PSe3 u1gy/HA jai1rta/FN_IGE_Me3 mint/KOT az/DET u1jon/IGE_PSe3_yedllett/IGE_TMe3 k1gyo1i/MN ./, amely/NM bo3re1t1/FN_PSe3_INS fe1l1ve1n1/FN_PSe3_SUP kanyarogva/HA keru2i1/IGE_Me3 minden/NM kavicsot/FN_FON ./.
E1s/KOT aki/NM tegnap/HA u1gy/HA halad1t/IGE_Me3 ce1lja1/FN_PSe3 fele1/NU ./, mint/KOT Krisha/IGE_Fe3 kilo3tt1/FN_PSe3 ny1la/FN_PSe3 ./, az/DET telto1va1z1va/HA meg-megell1t1t/IGE_Me3 most/HAA ,/, jobbra-bala/HAA ne1lze1v1/HAA ./, mint/KOT a/DET szerelmes/MN asszony/FN aki/NM elo3zo1so2r1/HAA ha1sz02k1k/IGE_e3 kedvese1hez/FN_PSe3כנהי/IGE_PSe3 DEL e1s/KOT la1bperceinek/FN_PSe3_DAT cso2rge1se/FN_PSe3 riasztja/IGE_Te3 az/DET e1j1/HAA csen1dben/FN_FON ./.
-/- O1h/FN Suryakanta/IGE_TMe3 ./, Suryakanta/IGE_TMe3 mi/NM lett/IGE_Me3 belo3led/NM ./ !/!
Mai1r/HAA este1edett/IGE_Me3 mikor/HAA a/DET palota/FN e1l1/NM e1rtek/IGE_e1 ./.
Mai1r/HAA a/DET nap/IGE_Me3 u1r1/IGE_Me3 valindoru1t1/HAA magat1/NM kialudt/IGE_Me3 ./.
Mai1r/HAA a/DET hold1/MN e1zu2stvize/FN_PSe3 csurgott/IGE_Me3 le/IK a/DET palota/FN malach1t1/FN_PSe3i ACC le1pcs031n/FN_PSe3i_SUP e1s/KOT a/DET ta1ncolo1/MN PA1va1k/FN_PL kapkodta1k1/FN_TMe3 la1bukat/FN_PSt3_INS ./, hogy/KOT meg/IK ne/HA a1zz1e1k1/IGE_TP1t1 ./.
Mai1r/HAA minden/NM fa1klya/FN kialud1t/IGE_Me3 csak/HAA a/DET legbelso3/MN szoba1ban/FN_FON e1g1y/IGE_Me3 e1gy/IGE_Fe3 illatozo1/MN Ila1bukat/FN_FON ./, mint/KOT a/DET so2t1/HAA palota1/FN_PSe3 DAT vila1gi1t1o1/MN mi1sz1ve/FN_PSe3 ./.
Balapandita/IGE_TMe3 virrasztott/IGE_Me3 Suryakanta/IGE_TMe3 kira1ly/FN arcke1rup/IGE_FON_PSe3 elo3t1t1/NM ./.
Mikor/HAA a/DET kira1ly/FN remego3/MN ke1zz1/FN_INS fe1r1lu1z1ta/FN_TMe3 a/DET fu2ggo2ny1/FN_FON_ACC e1s/KOT a/DET szoba1ba/FN_FON I1l1pett/IGE_Me3 volna/FN_Fe3 ./, a/DET bu2sz1ke/MN Balapandita/IGE_TMe3 felugrott/IGE_Me3 az/DET oroszla1n1bu1/MN Ila1bukat/FN_FON ./.
-/- K1/KOT vagy/KOT te/NM ./, e1s/KOT hagy/HAA KOT1 mersz1/FN_e2 a/DET kira1lyno3/MN szoba1lja1ba/FN_PSe3_DAT illepni/INF ?/?
-/- Kial1tt1a/IGE_TMe3 ha1trasz1ett/IGE_Me3 fejgel/FN_INS
haragos1n/NM_ESS_MOD ./.
-/- Suryakanta/IGE_TMe3 vagyok1/FN_e1 ./, a/DET fe1r1jed1/FN_TMe2 ./.
Nem/HA ismersz1/FN_e2 ?/?
-/- Szemtelen/NM csalo1/MN te/NM !/!
Suryakanta/IGE_TMe3 hangja/FN_PSe3 olyan/NM mint/KOT a/DET bagz01/MN tigris1/MN_FAC ./.
De/KOT a/DET te/NM hang1/FNarnings/IGE_Te2 mint/KOT a/DET here1/HAA FN_1CAU szolg1ke1/FN_POS u1gy/HAA remeg/FN ./.
Akkor/NM_TEM a/DET kira1ly/FN egy/DET le1pe1ssel/FN_INS ko2z1le1b1b1/MN le1p1tt/IGE_Me3 a/DET la1mpa1hoz/FN_ALL e1s/KOT i1gy/HAA KOT1z01/HAA_Ila1bukat/FN_FON ./.
Oh/NNP sze1pcsi1po3ju3/MN Balapandita/IGE_TMe3 ismerj/IGE_Pe2 meg/IK ./, mert/KOT e1n/NM vagyok/IGE_e1 a/DET fel1rjed/IGE_TMe2 Suryakanta/IGE_TMe3 kira1ly/FN ./.

Akkor/NM_TEM Balapandita/IGE_TMe3 egy/DET le1pe1st/FN_ACC haltrale1pet/IGE_Me3 e1s/KOT ilgy/KOT szol1t/IGE_Me3 ./: Suryakanta/IGE_TMe3 arca/FN_PSe3 mint/KOT a/DET Himalaja/FN_PS3 szikla1ja/FN_PSe3 keme1ny/MN e1s/KOT mozdulatlan/MN a/DET te/NM arcod/FN_PSe2 puha/MN e1s/KOT val1ttozol/MN mint/KOT a/DET sz11ne1szke1/IGE_TFe3 ./.

Jaj/ISZ !/

Oh/NNP Balapandita/IGE_TMe3 ./, Balapandita/IGE_TMe3 ismerj/IGE_Pe2 meg/IK !/

E1n/NM vagyok/IGE_e1 Suryakanta/IGE_TMe3 ./.

- Suryakanta/IGE_TMe3 kira1ly/FN ma/HA elindult/IGE_Me3 az/DET erdo3be/FN_ILL vezekelni/INF e1s/KOT Suryakanta/IGE_TMe3 zal1nde1ka1t/FN_PSe3_ACC soha/MN meg/IK nem/HA ma1si1tja/IGE_Te3 ./.

Jaj/ISZ !/

- O1h/FN Balapandita/IGE_TMe3 visszate1rtem/IGE_TMe1 hozza1d/NM_ALL ./, kert/KOT val1gyoldtam/IGE_TMe1 utalnad/NU ./.

- Suryakanta/IGE_TMe3 a/DET ce1l/FN ./, a/DET mozdulatlan/MN e1s/KOT ve1gso3/MN ./.

Nekem/FN kei/IGE_e0hzo3a1/MN_FAC mennem/IGE_INRe1 ./.

O3/NM nem/HA jo2n/IGE_e1hzo3a1n/FN soha/NM hajol1 ./.

Jaj/ISZ !/

- Eljo2ttem/IGE_TMe1 hozza1d/NM_ALL ./, hogy/KOT ta1masztod/IGE_TPe2 meg/IK gyo2kerezsa1kad/IGE_Me3 ingo1/MN lelkem/INF_FAC ./.

- Jaj/ISZ Suryakanta/IGE_TMe3 a/DET szikla/FN ./, melyhez/FN_ALL az/DET e1n/NM lelkem/FN_PSe1 ingo1/MN hajol1at/FN_PSe3 ACC ko2to2ttem/IGE_TMe1 ./.

Akkor/NM_TEM a/DET kira1ly/FN me1g/HA egy/DET le1pe1ssel/INF⛹FN_INS ko2zelebbb/MN le1pett/IGE_Me3 az/DET illatozol1/MN la1mpa1hoz/FN_ALL e1s/KOT ilgy/KOT szol1t/IGE_Me3 ./: - O1h/FN Balapandita/IGE_TMe3 ne1zz/FN_PSe2 ra1m/NM_SUB ./, ne1zzed/IGE_TMe2 a/DET kira1lyi/MN e1kszereket/FN_PL_ACC mellemen/INF_FAC ./.

E1n/NM vagyok/IGE_e1 a/DET fel1rjed/IGE_TMe2 Suryakanta/IGE_TMe3 ./.

Akkor/NM_TEM Balapandita/IGE_TMe3 me1g/HA e1s/KOT le1pe1ssel/INF⛹FN_INS haltrale1pet/IGE_Me3 e1s/KOT ilgy/KOT szol1t/IGE_Me3 ./: - Ha/KOT te/NM Suryakanta/IGE_TMe3 neve1t/FN_PSe3i ACC e1s/KOT kira1lyi/MN e1kszere1t/FN_PSe3i_ACC viseled/IGE_TMe2 akkor/HA Suryakanta/IGE_TMe3 helye1re/FN_PSe3_SUB le1pett1/IGE_Me2 ebben/NM_INE az/DET e1letben/FN_INE e1s/KOT Suryakanta/IGE_TMe3 kira1ly/FN ./, a/DET fel1rfitigris/FN=./, az/DET e1n/NM fe1rjemen/INF_FAC melhah/IGE_Me3 ./.

Ezt/NM_ACC mondta/IGE_TMe3 Balapandita/IGE_TMe3 e1s/KOT bu2szke/MN arca1t/FN_PSe3i_ACC kezeinek/FN_PSe3i_DAT lo1tusza/FN_PSe3 mo2ge1/NU rejte/HA si1rt/IGE_Me3 ./.

- O1h/FN Balapandita/IGE_TMe3 jai/IGE_TMe3 Pe2 nekem/NM_DAT !/

Te1ged/NM_ACC a/DET hu3se1g/FN_PSe3i_DAT hi1vnak/IGE_t3 e1s/KOT szerelmed/IGE_TMe2 a/llando1sala1gan/FN_INE volt/IGE_Me3 ingo1/MN lelkem/FN_PSe1 bizodalma/FN_PSe3 ./!

Akkor/NM_TEM Balapandita/IGE_TMe3 bu2szke1n/FN_PSe3_SUP kiegyenesedett/IGE_Me3 ./: - Bizony/HA hol1tomi/FN_TER hu3/MN maradok/FN_PL Suryakanta1hoz/FN_ALL kiniek/NM_DAT sz11vemet/FN_PSe1_ACC adtam/IGE_TMe1 ./.

Suryakanta1hoz/FN_ALL ki/IK a/DET fel1rfa1k/FN_PL ko2zt/NU olyan/NM volt/IGE_Me3 mint/KOT a/DET ko2vek/FN_PL ko2zt/NU az/DET adamanth/FN ./, de/KOT meghalt/IGE_Me3 ./.

135
Suryakanta hoz/NM_ALL akit/NM_ACC ez/NM az/DET arckel1p/FN albra1zol/IGE ,/, aki/NM Indral1to1l/FN_PL_ABL kapa/NM_TMe3 melto1salam1g1t/FN_PSe3_ACC ,/ a/DET tu3zto3l/FN_PL_ABL hev1t/FN_PSe3_ACC ,/ Jamato1l/FN_PL_ABL haragja1t/FN_PSe3_ACC ,/ e1s/KOT Ramato1l/FN_PL_ABL al1hatatossal1g1t/FN_PSe3_ACC ,/ de/KOT meghalt/IGE_Me3 ./.
Bizony/HA hu3/MN maradok/FN_PL hozza1/NM_ALL e1s/KOT el/IK nem/HA hagvom/FN emle1kek1t/FN_PSe3_ACC valaki1rt/FN_CAU akinek/NM_DAT arca/FN_PSe3 mint/KOT a/DET sz1nel1szek1/IGE_TFe3 ,/, hangja/FN_PSe3 mint/KOT a/DET here1lt/FN_CAU szolga1kek1/IGE.POS viselkede1se/FN_PSe3 mint/KOT a/DET terhes/ZN asszonyok1/IGE.POS olyan/NM ./.
Ezt/NM_ACC mondva1n/FN_PSe3_SUP felrehu1ztata/IGE_TMe3 az/DET ajo1t/FN arannyal/FN_INS hula1/IGE Me3 fu2ggo2nye1t/FN_PSe3_ACC e1s/KOT kikialltott/IGE_Me3 :/-/ Oh/NNP Cseti/IGE_Te3 ,/ kedes/MN csele1dem/FN e1bredj/IGE_Pe2 ./.
Fehe1t/MN ruha1t/FN_ACC hozz/IGE Pe2 nekem/NM_DAT e1s/KOT szeza1nmagokat/FN_PL_ACC ke1sz11ts/FN vil1zzel/FN_INS e1s/KOT puha/MN darbha-fu2vet/FN_ACC ./.
Akkor/NM_TEM Cseti/FN_PSe3 megjelent/IGE.Me3 az/DET aranyfu2ggo2ny/FN nyilla1sal1ban/FN_PSe3_INE e1s/KOT allomto1l/FN_ABL talgult/MN szemekkel/FN_PL_INS nelzett/IGE_Me3 a/DET szoba1ba/FN.ILL e1s/KOT halkan/MN.ESS.MOD szo1lt/Ite1/IGE_Me3 mint/KOT aki/NM fe1l/SZN ,/ hogy/KOT ve1lt/Ite1/IGE_Me3 aclam1bot1/FN_PSe3.ELA o2nmaga1t/NM.Acc le1bresztu/MN ./.
Q1h/FN bu1szze/MN Balapandita/IGE_TMe3 a/DET fehe1t/MN ruha/FN o2zvegyek/FN_PL ruha1ja/FN_PSe3 e1s/KOT szeza1nmagok/FN.PL puha/MN darbha/HU fu3vel/FN_INS halotti/MN a1ldozat/FN ./.
Mie1rt/HU k1val1n/Ite1 mindezt/NM.Acc ol1h/MN kirallyno3m/FN mikor/HU ott/HU la1tom/Ite1 a/DET fel1f1t/FN_ACC szolke1/IGE.Me3 nemle11n/FN_PSe3_SUP a/DET kira1lyi/MN e1kszereket/FN.PL ACC hordja/Ite1/IGE_Te3 me1g/HA a/DET test/FN de/KOT a/DET le1ket/FN.Acc belo3le/HA k1telpte/Ite1/IGE.TMe3 Yama/FN a/DET halal11/FN ura/FN_PSe3 ko2tel1hurok1/FN.PL_INS maga/MN uta1n/NU hu1zval1n/FN_PSe3_SUP ./.
Szo111tsd/Ite1/IGE_TPe2 szolgal1imat/FN_ACC kik/Ite1/IGE_T3 atyal1m/FN halza1bo11/FN_PSe3.ELA jo2tkek/Ite1/MMe3 velem/NM_INS ,/ hogy/KOT engem/NM.Acc hozza1/NM_ALL visszag1vezynek/MN.PL ./.
Atya1m/FN ke1sz11tsen/Ite1/IGE.Pe3 ma1gyalt/FN_PSe3. ACC szalmon1rda/FN_SUB ./., hogy/KOT a/DET tu3z/FN fu2st1vel/FN_PSe3.INS szal1lhassak/MN.PL a/DET meghalt/IGE. Me3 Suryakanta/IGE_TMe3 uta1n/NU ./, akihez/NM.ALL o2ro2kke1/HA hu3/MN maradok/FN.PL ./.