Information Extraction for Czech Based on Syntactic Analysis

Vít Baisa(✉) and Vojtěch Kovář

Masaryk University, Botanická 68a, 602 00 Brno, Czech Republic
{xbaisa,xkovar3}@fi.muni.cz

Abstract. We present a complex pipeline of natural language processing tools for Czech that performs extraction of basic facts presented in a text. The input for the tool is a plain text, the output contains verb and noun phrases with basic semantic classification. Automatic syntactic analysis of Czech plays a crucial role in the pipeline. In this paper, we describe the particular tools used in the system, then we give an example of its usage and conclude with a basic evaluation of the overall system accuracy.

1 Introduction

The term *information extraction* has been recently used in two main meanings. One of them is searching for documents relevant to a query in a large collection of documents. The other meaning is extracting simple facts from a text that can be further used e.g. for highlighting or indexing. In this paper, we will use the latter meaning of the term.

Current approaches to information extraction focus rather on restricted domains and are specialized in finding a particular type of information, e.g. named entities [1,2], relations between them [3] or protein interaction [4]. However, a generalized approach with high accuracy does not seem to be achievable at the moment.

In this article we describe a system designed as a pipeline of several language tools (including morphological tagging and disambiguation, syntactic analysis and semantic classification) used for extracting general information from a text, as opposed to the particular information extraction mentioned above. Currently, the tool gives a structured information about the facts found in the sentence and the result serves as an aid for further manual processing of extracted information. Its usage is not limited to any particular text type as the language tools used in the pipeline are robust enough to cover general language.

In the following chapters we introduce the particular language tools used in the system. Then we show an example of its usage and give a basic evaluation of the overall system accuracy.

2 System Overview

The system reads a text file with one sentence per line. In the first step we tokenize each sentence (one token per a line) according to spaces and a set of

Z. Vetulani and J. Mariani (Eds.): LTC 2011, LNAI 8387, pp. 155–165, 2014.
DOI: 10.1007/978-3-319-08958-4_13

punctuation. As a result we get a word-per-line format that is sometimes called *vertical* or *vertical text*.

This vertical file is then analysed by morphological analyser *Ajka* [5]. High amount of various morphological tags per token needs to be pruned – therefore we use morphological disambiguator *Desamb* [6].

Syntactic analysis, the main part of the system, follows: it identifies sentence constituents: noun, verb, prepositional and adjective phrases. We use the *SET* parser [7] with slightly modified output structure. The result of this step is a list of phrases together with the morphological information (lemmas and tags).

Simple rules in the next step classifies noun and pronoun phrases according to cases. Prepositional phrases are treated in a more complex way using hypernymic structure of *Czech WordNet* [8]. Thanks to the information from WordNet, we are able to classify the prepositional phrases as *manner*, *place*, *time* and more fine-grained classes, see below.

Resulting classification is clearly arranged in form of a table which is easily readable and suitable for further processing.

3 Morphological Analysis and Disambiguation

The very first step is tokenization based on simple splitting words by spaces and punctuation. Tokenization step outputs a text in vertical format (with word forms only) which serves as input for the next step, morphological analysis.

Due to rich morphology of Czech there are thousands of possible morphological tags. We used the attributive tagset built into the morphological analyser *Ajka* [5] where each tag encodes all relevant morphological information about a word-form: a part of speech (k), case (c), gender (g), numerus (n) etc.[1] The *Ajka* algorithm itself is based on matching the input words to a system of predefined model words that define the declension paradigm.

Some word-forms are heavily ambiguous – they may correspond to many tags as in example on Table 1. The example sentence *Pravidelné krmení je pro správný růst důležité* (*Regular feeding is important for a proper growth*) contains very high percentage of ambiguous words which is no exception in the Czech language. Note that even some of possible morphological tags were omitted in the example for the sake of brevity.

Once morphological analysis is done, the result must obviously be disambiguated for further syntactical analysis. For this purpose we use morphological disambiguator *Desamb* which is described more in detail in [6]. It uses combination of manually prepared and statistically learned disambiguational rules. For learning the statistics, the manually disambiguated DESAM corpus [9] was used. Apart from morphological disambiguation, the *Desamb* tool can also be used for detecting sentence boundaries.

An example result of such disambiguation can be seen on Table 1 in the third column. The fourth column contains linguistic interpretations of the morphological tags.

[1] For a full reference, see http://nlp.fi.muni.cz/projects/ajka/.

Table 1. Morphological analysis and disambiguation of a sample sentence.

word	**all possible tags**	**disambiguat.**	**interpretat.**
Pravidelné Regular	k2eAgMnPc4d1, k2eAgInPc1d1, k2eAgInPc4d1, k2eAgInPc5d1, k2eAgFnSc2d1, k2eAgFnSc3d1, k2eAgFnSc6d1, k2eAgFnPc1d1, k2eAgFnPc4d1, k2eAgFnPc5d1, k2eAgNnSc1d1, k2eAgNnSc4d1, k2eAgNnSc5d1, ... (*5 tags omitted*)	k2eAgNnSc1d1	adjective, singular, nominative case, neuter
krmení feeding	k2eAgMnPc1d1, k2eAgMnPc5d1, k1gNnSc1, k1gNnSc4, k1gNnSc5, k1gNnSc6, k1gNnSc3, k1gNnSc2, k1gNnPc2, k1gNnPc1, k1gNnPc4, k1gNnPc5	k1gNnSc1	noun, neuter, singular, nominative case
je is	k5eAaImIp3nS, k3p3gMnPc4, k3p3gInPc4, k3p3gNnSc4, k3p3gNnPc4, k3p3gFnPc4, k0	k5eAaImIp3nS	verb, third person, singular, present tense
pro for	k7c4	k7c4	preposition, accusative case
správný proper	k2eAgMnSc1d1, k2eAgMnSc5d1, k2eAgInSc1d1, k2eAgInSc4d1, k2eAgInSc5d1, ... (*18 tags omitted*)	k2eAgInSc4d1	adjective, accusative case, singular, masculinum inanimate
růst growth	k5eAaImF, k1gInSc1, k1gInSc4	k1gInSc4	noun, accusative case, singular, masculinum inanimate
důležité important	k2eAgMnPc4d1, k2eAgInPc1d1, k2eAgInPc4d1, k2eAgInPc5d1, k2eAgFnSc2d1, k2eAgFnSc3d1, k2eAgFnSc6d1, k2eAgFnPc1d1, k2eAgFnPc4d1, k2eAgFnPc5d1, k2eAgNnSc1d1, k2eAgNnSc4d1, k2eAgNnSc5d1, ... (*5 tags omitted*)	k2eAgNnSc1d1	adjective, nominative case, singular, neuter

4 Syntactic Analysis

The result of morphological analysis and disambiguation is then used as the input to the syntactic analyser SET [7]. This rule-based analyser based on the pattern matching principle offers a number of possible outputs, including dependency and phrasal trees, phrase extraction and others.[2]

The SET algorithm searches for all possible matches of the set of manually written rules. In the second phase, the best matches are selected to draw a syntactic tree. Only one tree is produced for each sentence.

Other output formats are produced by tree-traversing algorithms. For example, to output noun phrases the system searches the tree depth-first and if it finds a noun as a head of the phrase, it prints the corresponding sub-tree. Only the biggest subtrees (that represent the so-called maximal phrases) are output.

For our purposes the output in the form of noun, adjective, adverbial, prepositional and verb phrases seemed to be optimal. Using this output, we obtained a kind of predicate-argument structure where the verb phrase stands for the predicate and other phrases for its arguments.

For an example of the SET output, see Fig. 1.

5 Classification of Phrases

The final part of the pipeline consists of a classification of noun and prepositional phrases found by the syntactic analysis. To do this, we needed to create a set of labels for the information carried by particular phrases. They needed to be informative enough to be able to gain useful information from them; especially, they should represent some pieces of the information in the text that can be asked by a wh-question. On the other hand we wanted to design them to be not hard to detect so that the output is not distorted by a huge number of bad results.

In the current version of the system, the list of the labels on the boundary of syntax and semantics is in Table 2.

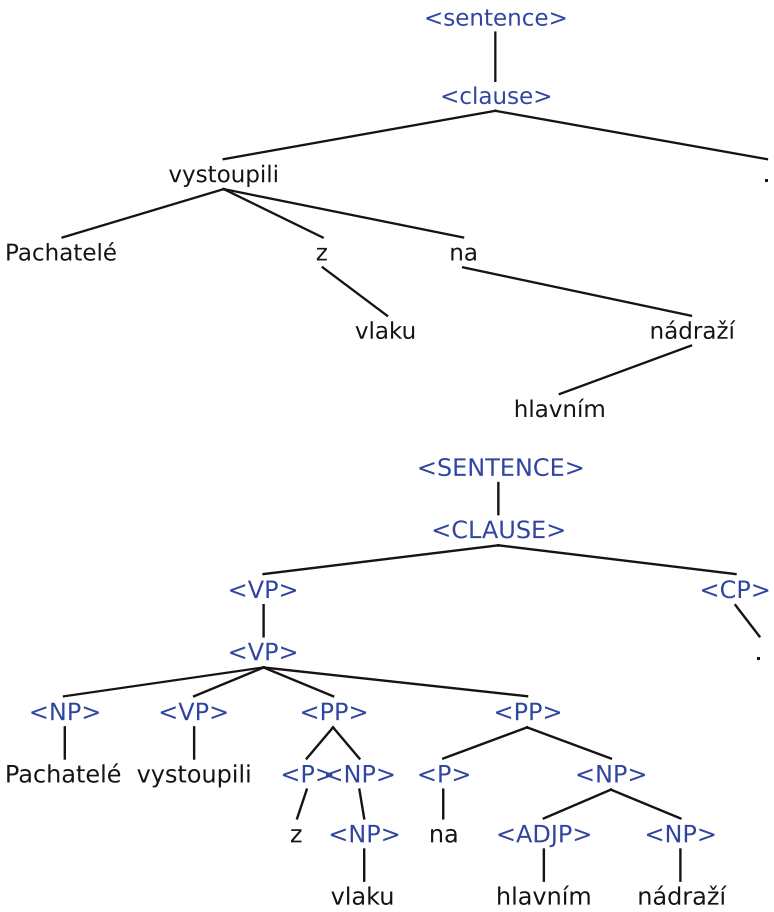
For classification of the noun phrases according to the classes described above we use a two-level rule-based system (as no annotated data in this format are currently available for Czech) with exploitation of the data available in the Czech WordNet [8].

5.1 Non-prepositional Phrases

At the first level, noun and adjective phrases obtained by the syntactic analysis are classified according to cases of phrase's headword.

In the current version of the system, straightforward rules are used: nominative case is always treated as SUBJECT. Genitive and dative cases are assigned for indirect object (INOBJ) and accusative case for direct object (DIROBJ)

[2] For a full reference, see http://nlp.fi.muni.cz/projects/set.




```
<s>Pachatelé vystoupili z vlaku na hlavním nádraží .
<clause>Pachatelé vystoupili z vlaku na hlavním nádraží
<vp> (): vystoupili
<phr> (k1c1gMnP): Pachatelé
<phr> (k7c2): z vlaku
<phr> (k7c6): na hlavním nádraží
</s>
```

Fig. 1. Example of the SET output: hybrid tree (on the left), constituent tree and phrasal output for sentence *Pachatelé vystoupili z vlaku na hlavním nádraží.* (*Perpetrators got off the train at the main station.*) Recognised phrases are (in the same order as in the example above): <vp> *got off*, <phr> *Perpetrators*, <phr> *the train*, <phr> *at the main station.*

Table 2. List of the labels used in the current version of the system.

SUBJECT	Answers to question *who* or *what*
VP	the Verb phrase representing the core of a clause; a predicate
INOBJ	Indirect object
DIROBJ	Direct object
MANNER	Prepositional phrase expressing manner that can be divided into subclasses WHY and HOW
TIME	a Time expression that can be further specified as SINCE-WHEN, TILL-WHEN or WHEN
LOC	Prepositional phrase expressing location that can be further divided into WHERE, FROM-WHERE, WHICHWAY and DIRECTION
ADDR	That corresponds to a phrase in vocative case; an addressee
INSTR	That corresponds to a phrase in instrumental case; an instrument

according to their functions in Czech. Vocative case corresponds with ADDR locative case is strictly prepositional in Czech and instrumental case is labeled with INSTR.

5.2 Prepositional Phrases

Classification of prepositional phrases is more complex as it can not be done simply by exploiting morphosyntactic features. For example, we can have *do Prahy* (to Prague) and *do hodiny* (within one hour), both having the same preposition and case but completely different meaning: the first one expresses location (covered by our LOC or more precisely DIRECTION), the other one is a time expression (covered by our TIME or more precisely TILL-WHEN). Clearly we needed to use some semantic information here.

Apart from the preposition itself and its case, we used the hyper-hyponymic hierarchy in the Czech WordNet together with heuristic rules for determining proper label – *time*, *location*, *manner* and their variants.

The classification algorithm then firstly prunes the space of the possible classifications based on the preposition and the case. There has been some investigation in classification of prepositions [10] but we decided to classify prepositions manually: this step is performed using rules transcribed from the Czech grammar handbook [11].

Then the algorithm searches WordNet hypernyms for all nouns in the phrase. If any of the pre-specified set of hypernyms is found, we are able to determine the phrase classification using a mapping of these hypernyms to the semantic classes. A sample of assigning WordNet hypernyms to our semantic classes can be found in Table 3.

Table 3. Example of assigning WordNet hypernyms to semantic classes.

Class	Example of hypernym
Time	Period, time, moment, dish, phenomenon, event
Location	Region, entity, group, state, continent
Manner	Organisation, quality, relationship

6 Examples and Application

In Table 4 there are some sample sentences together with extracted facts. As the motivation for development such kind of system was purely practical, in the very near future it will be employed as an aid for people analysing the text, in the form of highlighting, and as an intermediate step for text summarization.

Table 4. Examples of the output of the system.

Ministr vnitra Radek John včera v Praze schválil dlouho očekávaný zákon.				
SUBJECT	**VP**	**WHEN**	**DIROBJ**	**WHERE**
Ministr vnitra Radek John	schválil	včera	dlouho očekávaný zákon	v Praze
Minister of Inferior Radek John	has approved	yesterday	long awaited law	in Prague
Pachatelé z vlaku vystoupili až na hlavním nádraží.				
SUBJECT	**VP**	**FROM-WHERE**	**DIROBJ**	**WHERE**
Pachatelé	vystoupili	z vlaku		na hlavním nádraží
Perpetrators	got off	the train		at the main station
Při demonstracích byli zadrženi představitelé neonacistických spolků.				
SUBJECT		**VP**	**DIROBJ**	**WHERE**
představitelé neonacistických spolků		byli zadrženi		při demonstracích
neo-Nazi leagues representatives		were arrested		during demonstrations
Na semináři japonského šermu ve Valticích zemřel muž z Německa.				
SUBJECT	**VP**	**FROM**	**WHERE**	
muž	zemřel	z Německa	na semináři japonského šermu ve Valticích	
a man	died	from Germany	on japanese fencing seminar in Valtice	

Particularly, it will help the Czech police collaborators in the task of monitoring potentially dangerous internet discussion groups.

A web interface[3] of the tool was created for demonstration and for the purpose of its proper testing by real users. The interface allows users to input one sentence, a URL or an arbitrary text and performs the analysis, as described

[3] http://nlp.fi.muni.cz/projekty/set/efa/wwwefa.cgi/first_page

above. The output can be displayed in form of tables, XML or interactive highlighting mode similar to the GATE interface [12]. Screenshots of the interface are shown in Figs. 2, 3 and 4.

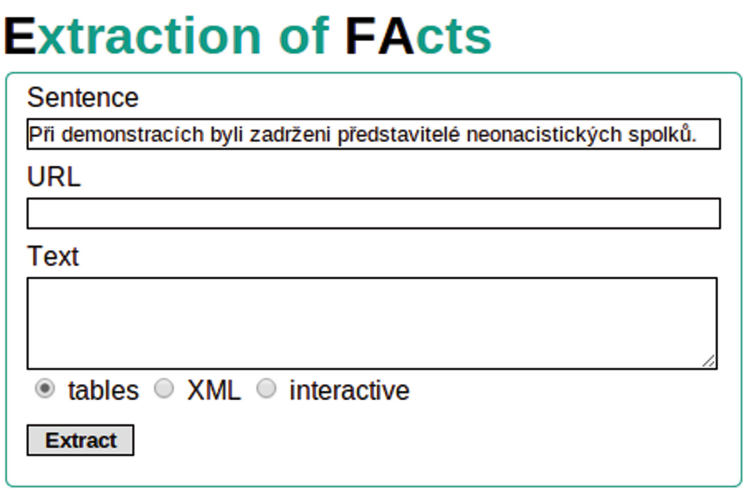


Fig. 2. Input form of the web interface for Extraction of FActs

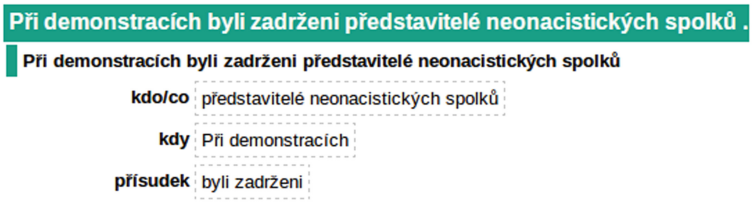


Fig. 3. Output in form of a table.

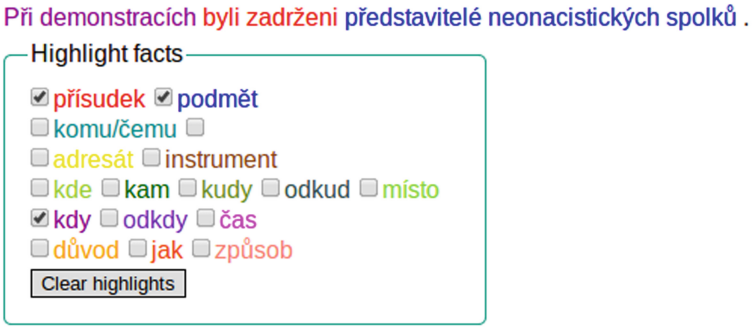


Fig. 4. Interactive output with highlighting facts according to their classification.

6.1 On Parsing Evaluation

In a more theoretical context, this work can contribute to the discussion related to the parsing evaluation problem [13], [14]. Most of the currently used parsing evaluation techniques are based on tree similarity metrics [15], [16] which is problematic with respect to the impact on usage of natural language parsers in real-world applications.

Miyao et al. [13] proposed another procedure of comparing results of various parsers based on measuring their contribution to a practically motivated task. We are strongly convinced that this is the right way in evaluating the quality of the parsers and that searching for more such evaluation applications (as an analogy to benchmark sets) is needed.

Above we have presented a system that exploits the results of the syntactic analysis and that may be adapted to different parsers. The task of the pipeline is well-defined and practically motivated. Furthermore, as opposed to the syntactic trees, the result of the analysis is very close to the information human beings are able to extract from the text. Therefore, judging correctness of the output would be simple and efficient.

7 Evaluation

The tools for tokenization, morphological analysis and disambiguation were not evaluated within the scope of this paper. Their evaluation can be found in [6,17].

Basic evaluation of phrase extraction and phrase evaluation was carried out manually on 50 randomly selected sentences from the internet news groups as bigger gold standard data for this task are unfortunately not available for the Czech language.

For the preliminary results performed on the small testing set see Table 5. We can see that the accuracy in phrase detection (F-1 measure in this case) is around 90 % and the accuracy of the classification (percentage of correctly classified phrases among the correctly detected phrases) reaches nearly 80 %. The overall accuracy of the pipeline is then around 70 % which can be interpreted as we get about 70 % of the text correctly identified and classified.

This is certainly not enough for tasks like automatic reasoning or precise question answering however it can be fairly useful in helping people read the texts faster, creating approximate summarizations or indexing based on the extracted information that could make searching the texts more efficient.

Table 5. Evaluation

Metric	Accuracy
Phrases detected properly	87.7 %
Phrases classified properly	79.7 %
Overall accuracy	**69.9 %**

8 Future Work

Since the system is designed as a pipeline, it is obvious that performance critically depends on performance of each component. In this respect we need to focus on all of these components in our future work.

For the first step – tokenization – we plan to improve the algorithm for finding boundaries of sentences (e.g. punctuation inside parentheses).

As for morphological analysis and disambiguation there is a room for developing better rules for the *Desamb* system or alternatively to skip this step completely and run the syntactic analysis on ambiguous input which is possible but currently performs worse than analysis with unambiguous input.

Phrase extraction should be enhanced with named entity recognition and improving maximal phrases detection namely in case of PP-attachment.

We also plan to refine the phrase classification with introducing more complex rules and semantic roles contained in the verb valency lexicon Verbalex [18].

The performance for the particular task of the Czech police can be also significantly improved by adapting the tools in the pipeline to the language of internet discussion groups that is slightly different from the general language, on all levels of description. This was not done in the first version of the system and the results may have been affected by this fact. As all of the tools in the pipeline are rule-based (so there is no need for text-type-specific annotated corpora) and their development was aimed at transparency, simplicity and scalability, it should be straightforward to adapt them to the special type of text and improve the overall accuracy of the system.

Also, as one of the future aims, we plan to adapt the task described in this paper to the needs of parsing evaluation. However, this will need more work, especially creating appropriate annotated data and adapting the pipeline to various parsers.

Acknowledgements. This work has been partly supported by the Ministry of the Interior of Czech Republic within the project VF20102014003 and by the Czech Science Foundation under the projects P401/10/0792 and 407/07/0679.

We would like to thank to all our colleagues which participated on developing used tools and data sources.

References

1. Etzioni, O., Cafarella, M., Downey, D., Popescu, A., Shaked, T., Soderland, S., Weld, D., Yates, A.: Unsupervised named-entity extraction from the web: an experimental study. Artif. Intell. **165**(1), 91–134 (2005)
2. Uchimoto, K., Ma, Q., Murata, M., Ozaku, H., Isahara, H.: Named entity extraction based on a maximum entropy model and transformation rules. In: Proceedings of the 38th Annual Meeting on Association for Computational Linguistics, Association for Computational Linguistics, pp. 326–335 (2000)
3. Hasegawa, T., Sekine, S., Grishman, R.: Discovering relations among named entities from large corpora. In: Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics, Association for Computational Linguistics (2004)

4. Abul Seoud, R.A., Youssef, A.B., Kadah, Y.M.: Extraction of protein interaction information from unstructured text using a link grammar parser. In: 2007 International Conference on Computer Engineering and Systems ICCES '07, Cairo, pp. 70–75 (2007)
5. Rychlý, P., Šmerk, P., Pala, K., Sedláček, R.: Morphological analyzer Ajka. Masaryk University, Technical report (2008)
6. Šmerk, P.: Unsupervised learning of rules for morphological disambiguation. In: Sojka, P., Kopeček, I., Pala, K. (eds.) TSD 2004. LNCS (LNAI), vol. 3206, pp. 211–216. Springer, Heidelberg (2004)
7. Kovář, V., Horák, A., Jakubíček, M.: Syntactic analysis as pattern matching: the SET parsing system. In: Proceedings of 4th Language and Technology Conference, Poznań, Poland, Wydawnictwo Poznańskie, pp. 978–983 (2009)
8. Pala, K., Smrž, P.: Building Czech WordNet. Rom. J. Inf. Sci. Technol. **7**(1–2), 79–88 (2004)
9. Pala, K., Rychlý, P., Smrž, P.: DESAM – annotated corpus for Czech. In: Jeffery, K. (ed.) SOFSEM 1997. LNCS, vol. 1338, pp. 523–530. Springer, Heidelberg (1997)
10. O'Hara, T., Wiebe, J.: Preposition semantic classification via penn treebank and framenet. In: Proceedings of the 7th Conference on Natural Language Learning at HLT-NAACL 2003-Vol. 4, Association for Computational Linguistics, pp. 79–86 (2003)
11. Karlík, P., Grepl, M., Nekula, M., Rusínová, Z.: Příruční mluvnice češtiny. Lidové noviny (1995)
12. Cunningham, H.: Gate: an architecture for development of robust hlt applications. In: Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL), pp. 168–175 (2002)
13. Miyao, Y., Sagae, K., Sætre, R., Matsuzaki, T., Tsujii, J.: Evaluating contributions of natural language parsers to protein-protein interaction extraction. Bioinformatics **25**(3), 394 (2009)
14. Jakubíček, M., Kovář, V., Grác, M.: Through low-cost annotation to reliable parsing evaluation. In: PACLIC 24 Proceedings of the 24th Pacific Asia Conference on Language, Information and Computation, Sendai, Japan, Tohoku University, pp. 555–562 (2010)
15. Harrison, P., Abney, S., Black, E., Flickinger, D., Gdaniec, C., Grishman, R., Hindle, D., Ingria, R., Marcus, M., Santorini, B., Strzalkowski, T.: Evaluating syntax performance of parser/grammars of English. In: Natural Language Processing Systems Evaluation Workshop: Final Technical report RL-TR-91-362, Griffiss Air Force Base, NY, Rome Laboratory, pp. 71–77 (1991)
16. Sampson, G.: A proposal for improving the measurement of parse accuracy. Int. J. Corpus Linguist. **5**(01), 53–68 (2000)
17. Sedláček, R., Smrž, P.: A new Czech morphological analyser ajka. In: Matoušek, V., Mautner, P., Mouček, C., Taušer, K. (eds.) TSD 2001. LNCS (LNAI), vol. 2166, pp. 100–107. Springer, Heidelberg (2001)
18. Hlaváčková, D., Horák, A.: Verbalex - new comprehensive lexicon of verb valencies for Czech. In: Proceedings of the Slovko Conference, Bratislava, Slovakia, VEDA (2005).