The Szeged Treebank

Dóra Csendes[1], János Csirik[1], Tibor Gyimóthy[1], András Kocsor[2]

[1] Department of Informatics, University of Szeged,
H-6720 Szeged, Árpád tér 2., Hungary
{dcsendes,csirik, gyimi}@inf.u-szeged.hu
[2] MTA-SZTE Research Group on Artificial Intelligence,
H-6720 Szeged, Aradi vértanúk tere 1., Hungary
{kocsor}@inf.u-szeged.hu

Abstract. The major aim of the Szeged Treebank project was to create a high-quality database of syntactic structures for Hungarian that can serve as a golden standard to further research in linguistics and computational language processing. The treebank currently contains full syntactic parsing of about 82,000 sentences, which is the result of accurate manual annotation. Current paper describes the linguistic theory as well as the actual method used in the annotation process. In addition, the application of the treebank for the training of automated syntactic parsers is also presented.

1 Introduction

The availability of accurately annotated data is becoming an increasingly important factor in the developments of Computational Linguistics. To support that, linguists and developers of natural language processing systems design different annotation schemes and tools which allow for adding as much linguistic information to texts as possible. Inspired by the research results of the Penn Treebank and several other treebank projects, our research group set out to create a golden standard treebank for Hungarian, containing reliable syntactic annotation of texts. Project work contained the selection and adjustment of the theory used for syntactic analysis, the design of the annotation methodology, the adaptation of the available tag-sets to Hungarian, automated pre-processing, manual validation and correction, and experiments with machine learning methods for automated parsing. The current paper presents an overview of the Szeged Treebank initiative and its results to date.

The treebank currently contains detailed syntactic analysis of approx. 82,000 sentences (1.2 million words) based on a generative grammar approach. Annotated files are available in XML format using the TEI DTD P2[1] scheme. Ideally, the treebank should contain samples of all the syntactic structures of the language, therefore, it serves as a reference for future corpus and treebank developments, grammar extraction and other linguistic research. It also serves as a

[1] The TEI DTD description is available at the following website: http://www.tei-c.org

reliable test suite for different NLP applications, as well as a basis for the development of computational methods for both shallow and deep syntactic parsing, and information extraction. Well-defined methods or elaborate theoretical foundations for the automated syntactic analysis of Hungarian texts were lacking at the start of the project. For this reason, novelty of the project work lies in the design of a practical approach for syntactic annotation of natural language sentences.

The current paper is structured as follows. After commenting on related treebank initiatives for other languages, we continue to introduce the backgrounds of the project and the theory designed for the syntactic annotation of texts. In section 4, we describe the used tag-set and the annotation process in some detail, while in section 5 we discuss results achieved by machine learning algorithms for automated syntactic parsing of texts. We close the paper with some words about current and future works.

2 Related works

Treebanks typically contain morphological, morpho-syntactic, syntactic and sometimes even semantic information about a language, therefore, they are a valuable source of further research in the fields of theoretical linguistic and computational language processing. Treebanks – especially if manually annotated – greatly help the development of effective syntactic parsers and other automated tools used for the analysis of natural language. Because of their efficient applicability in computational linguistics, numerous treebank projects have been initiated over the past ten years.

One of the most notable of all, the Penn Treebank project [12] produced skeletal parses on top of an initial POS tagging showing rough syntactic and semantic information on about 2.5 million words of American English. Syntactic parsing of the texts included the annotation of predicate-argument structures.

Another prominent treebank proposition is the Prague Dependency Treebank (PDT) [7] for Czech. The project's theoretical background is a dependency-based syntax handling the sentence structure as concentrated around the verb and its valency, but also containing the dimension of coordination. Texts are annotated on the morphological, syntactic and the tectogrammatical (linguistic meaning) levels, therefore, the nodes of the dependency tree are labelled by symbols containing information about all three of these levels. An attempt to incorporate information on discourse structure (topic-focus opposition) has also been initiated by researchers of the PDT project.

Several other projects for Slavic languages follow the PDT approach. The Slovene Dependency Treebank (in progress), for example, aims to add syntactic annotation to the available morphologically annotated TELRI corpus using analytic tree structures. In the Dependency Treebank for Russian [3], the syntactic data are also expressed in the dependency formalism, but the inventory of syntactic functional relations is considerably richer than in the PDT. With its unique approach of HPSG-based annotation, the BulTreeBank [14] is an excep-

tion. It contains detailed syntactic structure for 1 million words in a graph-form following the HPSG scheme which allows for a consistent description of linguistic facts on every linguistic level, incl. phonetic, morphological, syntactic, semantic and discourse.

The TIGER Treebank [5] is a more recent initiative for German language. Its first version contains 35,000 syntactically annotated sentences from the Frankfurter Rundschau newspaper, but the project intends to build the largest and most exhaustively annotated natural language resource for German. In its encoding, the TIGER Treebank uses a hybrid combination of dependency grammar and phrase structure grammar. The Turin University Treebank (TUT) [10] built for Italian combines the dependency approach with the predicate-argument structure paradigm of the PennTreebank project and is characterized by a rich grammatical relations system.

Some treebank annotation schemes aim at theory-independent interpretation, like the ones applied in the Spanish Treebank [16] or the French Treebank [1]. Treebank projects are also initiated for several other languages, such as Swedish [13], Japanese (the Hinoki Treebank) [4], Turkish [2], Arabic [11], just to mention a few.

3 Preliminaries and theoretical guidelines

3.1 Szeged Corpus as the predecessor

The Szeged Treebank project was preceded by an extensive, four-year-long work aimed at the creation of a golden standard corpus for Hungarian language. The resulting Szeged Corpus is a manually annotated natural language database comprising 1.2 million word entries (with 145,000 different word forms) and an additional 225,000 punctuation marks [6]. With this, it is the largest manually processed Hungarian textual database that serves as a reference material for corpus linguistic research and applications for the language. It is a thematically representative database containing texts from six different genres, namely: fiction, newspaper articles, computation-related scientific texts, short essays of 14-16-year-old students, legal texts, and short business news.

Language processing of the Szeged Corpus includes morphological analysis, POS tagging and shallow syntactic parsing. Shallow parsing went as far as marking bottom-level NP structures, and clause annotation. Machine learning methods for POS tagging [9] and shallow parsing [8] have been trained on the corpus with considerable success. High accuracy results (over 97% per word accuracy) for POS tagging are especially notable, considering the fact that (i) the richness of Hungarian morphology poses a considerable challenge to automated methods, and that (ii) due to the applied encoding scheme, the ratio of ambiguous words are almost 50%.

3.2 Theoretical background, methodology and new approaches

Since no syntactic annotation schemes were available for Hungarian, the major challenge of the Szeged Treebank project was to adapt the theoretical founda-

tions of Hungarian syntax to a more practical syntactic annotation methodology. When designing the methodology, researchers aimed to:

- demonstrate the varieties of Hungarian syntactic patterns exhaustively;
- stay in correlation with the newest linguistic theories[2];
- create an annotation scheme that can be used extensively in later research activities and in computer assisted practical solutions.

Research results showed that the most promising theoretical frame for the definition of the annotation scheme would be generative syntax in combination with certain dependency formalism, (the latter being considered more suitable for languages with free word order). Our approach resembles dependency-based syntax to the extent in which it handles the sentence structure as concentrated around the verb and its argument structure, but it does not assign syntactic types to each sentence component relation. However, the proposed structure does contain information as to which components of the sentence are syntactically linked, and describes each node of the tree with complex labels. These labels contain morphological and syntactic description of the sentence components in the form of attributes.

In building a syntactic tree, the initial step is the (re)creation of the deep sentence structure. In a deep structure of a Hungarian sentence, it is always the verb that stands in the first position and it is followed by its arguments. Since Hungarian has a relatively free word order, arguments of the verb can move anywhere in the sentence occupying so-called functional positions. Naturally, by moving certain arguments, the meaning of the sentence is likely to change accordingly. Arguments that moved somewhere else, leave traces in their original position, which are indexed to their newly occupied position (see Figure 1., i, j, k elements). When applying this theory to the Szeged Treebank's XML format, we decided not to keep the traces in the treebank, instead, we added a new NODE label within the verb phrase and described the given argument with attributes. The resulting syntactic trees do not appear in the form of a tree, but as bracketed XML structures, (however, the transformation into a tree is always possible). The first figure (Fig. 1) shows the original tree with the argument traces, while the second one (Fig. 2) illustrates our XML representation of the same sentence.

The features of the defined treebank annotation formalism allows for the description of particular linguistic structures and phenomena occurring in Hungarian. It organises the represented information in different layers, keeping them separate to facilitate the selection of data during a number of large-scale NLP applications incl. information extraction, phrase identification in information retrieval, named entity recognition, machine translation, and a variety of text-mining operations.

[2] References: É. Kiss K., Kiefer F., Siptár J.: *Új magyar nyelvtan*, Osiris Kiadó, Bp., 1999.; Alberti G., Medve A.: *Generatív grammatikai gyakorlókönyv I-II.*, Janus/Books, Bp., 2002.; Kiefer F., ed.: *Strukturális magyar nyelvtan I. Mondattan*, Akadémiai Kiadó, Bp., 1992.

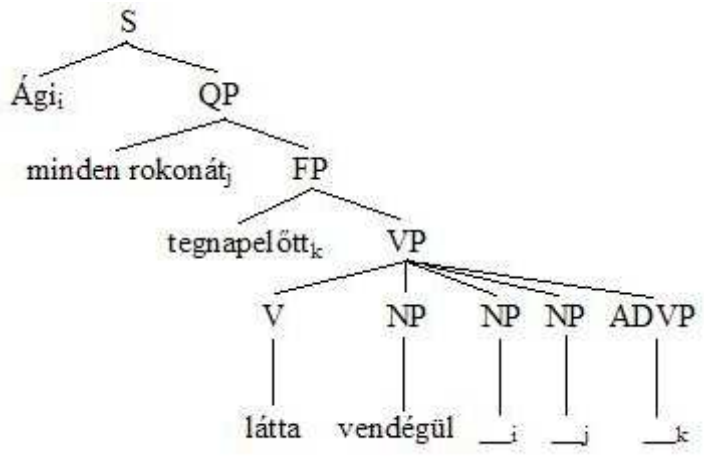

Fig. 1. A syntactic tree example.

```
<CP id="file.1.1">
<NP id="file.1.2"> Ági </NP>
<NP id="file.1.3">
<ADJP> minden </ADJP>
rokonát
< /NP>
<ADVP id="file.1.4"> tegnapeltt </ADVP>
<V_ id="file.1.5">
<V0> látta </V0>
<CHILDREN>
<NODE idref="file.1.2" type="NP" role="NOM">  </NODE>
<NODE idref="file.1.3" type="NP" role="ACC">  </NODE>
<NODE idref="file.1.4" type="ADVP" role="TLOCY">  </NODE>
<NODE idref="file.1.6" type="NP" role="ESS">  </NODE>
< /CHILDREN>
< /V_>
<NP id="file.1.6"> vendégül </NP>
<c> . </c>
< /CP>
```

Fig. 2. A syntactic tree example using XML.

4 Annotation of the Szeged Treebank

Similarly to the majority of annotation projects, the Szeged Treebank also follows the Penn Treebank approach, which distinguishes an automatic annotation step followed by manual validation and correction.

4.1 The set of syntactic tags

The tag-set used in the project shows correlation with many other internationally accepted syntactic tag-sets. The list of tags is as follows:

- ADJP: adjectival phrases
- ADVP: adverbial phrases, adverbial adjectives, postpositional personal pronouns
- c: punctuation mark
- C0: conjunctions
- CP: clauses (also for marking sentences)
- INF_: infinitives (INF0, CHILDREN, NODE)
- NEG: negation
- NP: noun phrases (groups with noun or predicative adjective or inflected personal pronouns as head)
- PA_: adverbial participles (PA0, CHILDREN, NODE)
- PP: postpositional phrases
- PREVERB: preverbs
- V_: verb (V0, CHILDREN, NODE)
- XP: any circumstantial or parenthetic clause that is not a direct part of the sentence

Attributes of a node may contain information about the node's type (e.g., NP, ADVP, etc.), and its morpho-semantic role in the sentence (e.g., nominative, instrumental, inessive, terminative, locative, etc.) also to be seen in Figure 2.

4.2 Pre-processing of the texts

Pre-processing of the texts was conducted in two steps. Initially, the full structure of NPs was marked. Since Hungarian is a highly inflectional language, the grammatical role of a certain syntactic unit is typically defined by the inflection of its head. Due to the fact that it is mostly NPs that occur as heads of a syntactic unit, it can be said that the grammatical structure of Hungarian sentences are determined by inflected nominal structures, therefore, it was crucial to mark NPs in the first phase. Automatic pre-parsing of the sentences was completed with the help of the CLaRK[3] program, in which syntactic rules have been defined by Hungarian linguistic experts for the recognition of NPs. The basic mechanism of CLaRK for linguistic processing of text corpora is a cascaded regular grammar processor. The manually defined NP annotation rules heavily rely on the use of such regular expressions that, in turn, define increasingly complex NP structures. Initially, base NPs containing noun heads are identified. Following that, more complex NP structures are defined based on the coordination and/or merge of possessive NPs and NPs with heads other than nouns. Rules can be applied to the same piece of text recursively. A remarkable ∼70% accuracy was already

[3] The CLaRK system was developed by Kiril Simov at the Bulgarian Academy of Sciences in the framework of the BulTreeBank project (http://www.bultreebank.org).

achieved in the pre-parsing phase, due to the efficient definition of expert rules. For the pre-parsing of all other structures (ADJP, ADVP, etc.), we developed our own tool, which applies manually defined simple grammatical rules for the automated pre-annotation of sentences.

4.3 Manual validation of syntactic trees

Manual validation and correction of the syntactic structures and their attributes was performed by a group of linguist especially trained for this task. They used a locally developed editor for the task and worked 24 person-months on the project.

Considering the annotation of the inner structure of NPs, certain difficulties have to be highlighted. Firstly, it should be noted that marking the boundaries (beginning and ending) of NPs is a problematic matter, the reason for which is the possible replacement of a noun head of the NP with its modifiers. Another problematic area is the left recursive insertion of progressive and perfect participles that often bring several adjuncts (sometimes long embedded clauses) into the NP. To avoid problems deriving from such peculiarities of Hungarian language, carefully defined rules were laid down for the manual correction of NPs. Due to the lack of space, we will not describe the manual validation of other syntactic constituents in detail, but it has to be noted that it proved to be much more straight-forward than that of the NPs. As a result of the annotation, we receive the detailed structure of the syntactic tree and the functional description of every node.

5 Training and testing machine learning algorithms for full syntactic parsing

Textual data is often used for the training of different machine learning methods in order that they can solve problems occurring in the field of Computational Linguistics. While there are several methods that use text in its raw, unanalysed form (cv. unsupervised training), more accurate results can be obtained by using annotated corpora for the training.

Research groups studying the structure of Hungarian sentences have made a great effort to produce a consistent and extensive syntax rule system, yet these are not or just partially adapted to practical, computer related purposes so far. This implied that there is a strong need for a technology that would be able to divide a Hungarian sentence into syntactical segments, recognize their structure, and based on this recognition, would assign an annotated tree representation to each sentence. The main goal, therefore, was to develop a generally applicable syntactic parser for Hungarian based on the Szeged Treebank annotations. Different learning methods have been studied, such as rule-based, numeric and logic algorithms. Taking into consideration the specific features of Hungarian language, it was found that logic methods can be best applied to our puroses, therefore a parser was developed based on this founding.

For training and testing the parsers, we used a set of 9600 sentences (thematically selected from the business news domain) divided into 10 sections for ten-fold cross validation. The input of the parsers was morphologically analysed text and the output was bracketed syntactically analysed sentences. Parsing rules were retrieved from the annotated Szeged Corpus and were combined with manually defined ones.

5.1 NP recognition

The table below shows average results of the ten-fold cross validation test performed by the developed parser for the recognition of NPs.

Categories of recognition	**Precision**	**Recall**	**Fβ=1**
Complete NP structures	81.28%	87.43%	84.24%
Boundaries (first and last elements) of NP structures	88.31%	92.08%	90.15%
NP structures (depth<=2)	86.02%	89.72%	87.83%
NP structures (depth>2)	74.71%	78.19%	76.41%

Table 1. NP recognition results.

5.2 Full syntactic parsing

In the case of full syntactic parsing, we aimed at the recognition of shorter multi-level tree structures, incl. ADJPs, ADVPs, PAs, etc. The training resulted in ∼1500 different tree patterns where the leaves contain detailed morphological and morpho-semantic information about the component. Test results for full parsing of short trees can be seen in the following table.

A0 to A9 are the ten sections of the treebank that were selected for the training of the parser. Columns 'Yes' and 'No' show whether the parser's guess about a certain structure was correct or not (i.e., whether it recognises a structure as a syntactic one, and if yes, what kind). 'Accuracy' measures were calculated from these results. The 'Etalon' column presents the number of manually marked syntactic structures, thus, the golden standard. The 'Predict' column shows the number of structures that were identified by the parser, while the 'Correct' column shows the number of correctly identified structures.

Results of Table 2. are only preliminary ones, and can be considered as baseline results in syntactic parsing of Hungarian sentences. It must be admitted that better results are already available for other languages (cf. results of the Link, NLTK, Stanford Parser, Apple Pie parsers), but due to the fact that this is a fresh initiative for Hungarian, and that the number of tree patterns is much higher than for other languages, results can be considered promising. Further improvements in this field are the nearest future plan of the group.

	Classification			**Tree pattern recognition**					
	Yes	No	Accuracy	Etalon	Predict	Correct	Precision	Recall	Fβ=1
A0	12688	2411	84,03%	5978	5648	4341	76,86%	72,62%	74,68%
A1	11788	2704	81,34%	6291	5595	4350	77,75%	69,15%	73,20%
A2	12476	2619	82,65%	6390	5733	4486	78,25%	70,20%	74,01%
A3	11835	2471	82,73%	6097	5419	4326	79,83%	70,95%	75,13%
A4	11031	1607	87,28%	5347	5286	4398	83,20%	82,25%	82,72%
A5	11740	1585	88,11%	5577	5553	4677	84,22%	83,86%	84,04%
A6	11404	1622	87,55%	5488	5440	4562	83,86%	83,13%	83,49%
A7	11624	1596	87,93%	5640	5489	4656	84,82%	82,55%	83,67%
A8	12052	2079	85,29%	5989	5739	4676	81,48%	78,08%	79,74%
A9	12499	2811	81,64%	6691	5755	4593	79,81%	68,64%	73,81%
Average			***84,85%***				***81,01%***	***76,14%***	***78,45%***

Table 2. Recognition results for full syntactic structures.

6 Current and future works

As a first step, we intend to improve the results of automated syntactic parsing both on the shallow and the detailed levels. With sufficiently reliable parsers, we will be able to create larger databases, and improve our information extraction (IE) system as well. Current results are already implemented in the IE system, and preliminary tests indicate that results are better than that achieved with shallow parsing, therefore, there is a good chance for further improvement. To support IE from another perspective, some of our current work aims at building general top-level and detailed domain specific ontologies for Hungarian. As a continuation of the Szeged Treebank project, we intend to enrich the texts with detailed semantic information in the future. Using the results of previous and future projects, we aim at developing a fully automated method for the extensive analysis and processing of Hungarian sentences on all levels.

References

1. Abeillé, A., Clément, L., Toussenel, F.: *Building a Treebank for French* in A. Abeillé (ed) *Treebank:. Building and Using Parsed Corpora*, Kluwer Academic Publishers, pp 165-187 (2003)
2. Atalay, N.B., Oflazer, K., Say, B.: *The Annotation Process in the Turkish Treebank* in Proceedings of the EACL'03 Workshop on Linguistically Interpreted Corpora (LINC), Budapest, Hungary (2003)
3. Boguslavsky, I., Grigorieva, S., Grigoriev, N., Kreidlin, L., and Frid, N.: *Dependency treebank for Russian: concepts, tools, types of information* in Proceedings of COLING-2000, Saarbrücken, Germany (2000)
4. Bond, F., Sanae F., Chikara H., Kaname K., Shigeko N., Nichols, E., Akira O., Takaaki T., Shigeaki A.: *The Hinoki Treebank: A Treebank for Text Understanding* in Proceedings of the IJCNLP 2004, Hainan Island, China and in LNCS vol. 3248 (2004)

5. Brants, S., Dipper, S., Hansen, S., Lezius, W. and Smith, G.: *The TIGER Treebank* in Proceedings of the Workshop on Treebanks and Linguistic Theories (TLT 2002), Sozopol, Bulgaria (2002)
6. Csendes, D., Csirik, J., Gyimóthy, T.: *The Szeged Corpus: A POS tagged and Syntactically Annotated Hungarian Natural Language Corpus* in Proceedings of TSD 2004, Brno, Czech Republic and LNAI vol. 3206 (2004)
7. Hajic, J.: *Building a Syntactically Annotated Corpus: The Prague Dependency Treebank* in Issues of Valency and Meaning, pp. 106-132, Charles University Press, Prague (1999)
8. Hócza, A., Iván, Sz.: *Learning and recognizing noun phrases* in Proceedings of the Hungarian Computational Linguistics Conference (MSZNY 2003), pp. 72-79, Szeged, Hungary (2003)
9. Kuba, A., Csirik, J., Hócza, A.: *POS tagging of Hungarian with combined statistical and rule-based methods* in Proceedings of TSD 2004, Brno, Czech Republic and LNAI vol. 3206 (2004)
10. Lesmo, L., Lombardo, V., Bosco, C.: *Treebank Development: the TUT Approach* in Proceedings of ICON 2002, Mumbay, India (2002)
11. Maamouri, M., Bies, A., Buckwalter, T., Mekki, W.: *The Penn Arabic Treebank: Building a Large-Scale Annotated Arabic Corpus* in Proceedings of the NEMLAR International Conference on Arabic Language Resources and Tools, Cairo, Egypt, (2004)
12. Marcus, M., Santorini, B., Marcinkiewicz, M.: *Building a large annotated corpus of English: the Penn Treebank* in Computational Linguistics, vol. 19 (1993)
13. Nivre, J.: *What kinds of trees grow in Swedish soil? A comparison of four annotation schemes for Swedish* in Proceedings of the Workshop on Treebanks and Linguistic Theories (TLT 2002), Sozopol, Bulgaria (2002)
14. Osenova, P., Simov, K.: *BTB-TR05: BulTreeBank Stylebook*, BulTreeBank Project Technical Report 05 (2004)
15. Simov, K., Simov, A., Kouylekov, M., Ivanova, K., Grigorov, I., Ganev, H.: *Development of Corpora within the CLaRK System: The BulTreeBank Project Experience* in Proceedings of the Demo Sessions of EACL'03, pp. 243-246, Budapest, Hungary (2003)
16. Torruella, M.C., Antonín, M.: *Design Principles for a Spanish Treebank* in Proceedings of The Workshop on Treebanks and Linguistic Theories (TLT2002), Sozopol, Bulgaria (2002)