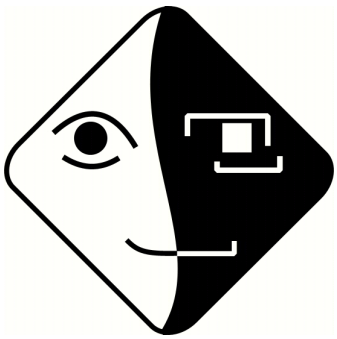
The Szeged Treebank Project



University of Szeged, Department of Informatics

Human Language Technology Group

www.inf.u-szeged.hu/hlt

Project partners:

MorphoLogic Ltd. Budapest

Research Institute for Linguistics at the Hungarian Academy of Sciences

Motivation

To create a **high-quality and reliable database of full syntactic structures** for Hungarian language that can serve as a golden standard to further research in linguistics and computational language processing.

Main aims:

- to demonstrate the varieties of Hungarian syntactic patterns exhaustively;
- to stay in correlation with newest linguistic theories;
- to create a methodology and an annotation scheme that can be used in later research activities;
- to experiment with machine learning algorithms for full syntactic parsing.

General Information about the Szeged Treebank

Size of the treebank:

- **82 thousand sentences**
- **1.2 million word entries**

Texts of the treebank:

The Szeged Treebank was based on the Szeged Corpus, which is a representative collection of texts deriving from 6 different genres:

- **Fiction (3 novels)** representing special vocabulary and often complicated or unusual phrase structures.
- **Short essays** of 14-16-year-old students. These texts contain shorter, less complicated sentences, but misspelled words and/or grammatical mistakes occur often in them.
- **Newspaper articles** (from 3 daily and 1 weekly paper).
- **Texts related to computer science**
- **Legal texts** coming from Hungarian laws passed on economic enterprises and authors' rights. Sentences are typically very long, irregularly structured, often fragmented, and are full of cross-references. For this reason, legal texts proved to be the most difficult to handle by automated methods and to prepare for language technology developments.
- **Short business news**

Syntactic analysis and annotation

Theoretical background

Tasks carried out:

- Selection and adjustment of the theory used for syntactic analysis: **generative syntax in combination with certain dependency formalism** has been used.

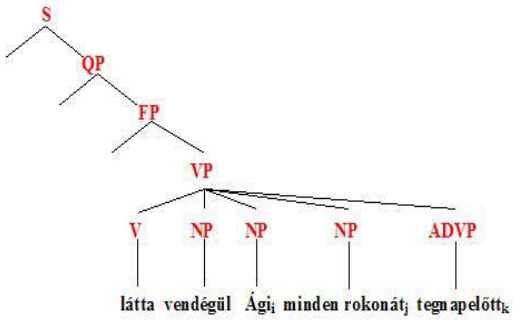


Figure 1. Before

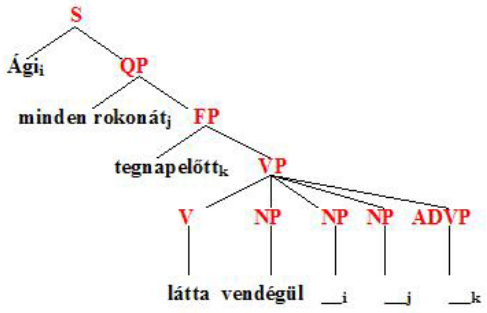


Figure 2. After

- Design of the annotation methodology: the description of nodes are carried out with using complex **labels that contain morphological and syntactic information about the sentence components in the form of attributes.** The methodology correlates with international XML standards.

```
<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"> tegnapelőtt </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>
```

Figure 3. XML version

Annotation of the Szeged Treebank

Adaptation of the available tag sets to Hungarian:

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

Automated pre-analysis

Automated pre-parsing for NPs was completed with the help of the CLaRK program[1], in which regular rules have been defined by linguistic experts for the recognition of NPs. The coverage of the rules defined in CLaRK was ~70%. Example of an NP recognition rule defined in CLaRK:

S_NP1

a) ***<"[T#">,(<"[Pi@@@n#">|<"[Pg@@@n#">|<"[M#">)*, (<"[R#">*,(<AP>|<"[A#">|<"[Ps@@@n#">))*,<"[N#">***

b) ***<"[T#">,(<AP>|<"[A#">),<"[M#">+,<"[N#">***

The automated pre-annotation of all other syntactic structures was competed with a software developed by the group.

Manual annotation

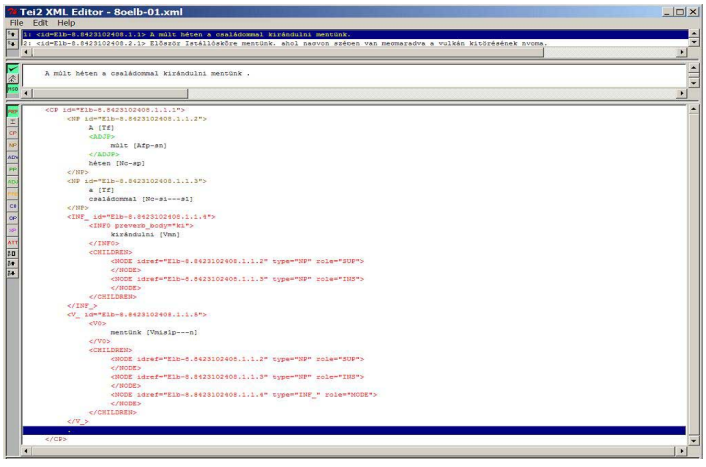


Figure 4. Syntax editor

[1] 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).

Training and testing machine learning algorithms for full syntactic parsing

Different learning methods have been used:

- C4.5 rule-based method
- SVM numeric method
- PGS logic algorithm developed by members of the group

For training and testing the parsers, a set of 9600 sentences was used that had been divided into 10 sections for **ten-fold cross validation.**

Input: morphologically analysed simple text.

Output: bracketed syntactically analysed sentences in XML.

A	baseline	C4.5	PGS	SVM
Accuracy	57,70%	85,99%	84,85%	**86,24%**
	1,11%	*2,75%*	*2,71%*	***0,35%***
Precision	-	**85,17%**	81,01%	81,88%
	-	*2,54%*	*2,91%*	***0,19%***
Recall	-	75,08%	76,14%	**76,82%**
	-	*6,22%*	*6,41%*	***0,57%***
F-value	-	**79,75%**	78,45%	79,27%
	-	*4,59%*	*4,69%*	***0,26%***

Table 1. Recognition results for full syntactic structures

In general, the **SVM algorithm has some advantage** over the other methods (best results are bolded).

The results illustrated above are only preliminary ones, and can be considered as **base-line results in syntactic parsing of Hungarian sentences**.

Current and future work

- Development of a Hungarian-English **machine translation** technology
- Building **WordNet** for Hungarian; research on **ontology** building methodologies for Hungarian; building domain specific and general ontologies for Hungarian
- Developing **automated methods** extensive **semantic** analysis and processing of Hungarian texts