BTB-TR01: BulTreeBank Project Overview*

Kiril Simov
BulTreeBank Project
http://www.BulTreeBank.org
Linguistic Modelling Laboratory, Bulgarian Academy of Sciences
Acad. G. Bonchev St. 25A, 1113 Sofia, Bulgaria
kivs@bultreebank.org

BulTreeBank Technical Report BTB-TR01

31.08.2004

Abstract

The BulTreeBank Project aims at the creation of a detailed syntactic treebank of Bulgarian. The annotation scheme of the treebank is based on the HPSG linguistic theory.

1 Introduction

The BulTreeBank project's main goal is the creation of a high quality syntactic treebank of Bulgarian which is HPSG oriented. In the course of reaching this main goal we had to produce additional language resources and tools which lacked for Bulgarian. These results of the project can be summarized as follows:

- A set of Bulgarian sentences marked-up with detailed syntactic information. These sentences are mainly extracted from authentic Bulgarian texts. They are chosen with respect to two criteria. First, they cover the variety of syntactic structures in Bulgarian. Second, they show the statistical distribution of these phenomena in real texts.
- Inside the Treebank a core set of sentences are designated. The purpose of this core set of sentences is to serve as a test-suite for software application processing texts in Bulgarian at the level of syntactic descriptions.
- Reliable partial grammar for automatic parsing of phrases in Bulgarian. This grammar is extensively tested and used during the creation of the Treebank. It will be used as a module separate from the Treebank in tasks which require only partial parsing of natural language texts, such as Information Retrieval, Information Extraction, Data Mining from Texts and etc.
- Software modules for compiling, manipulating and exploring the data in the Treebank. This software supports both: the creation of the Treebank, and its usage for different purposes, such as automatic extraction of grammars for Bulgarian.

*The work reported here is done within the BulTreeBank project. The project is funded by the Volkswagen Stiftung, Federal Republic of Germany under the Programme *"Cooperation with Natural and Engineering Scientists in Central and Eastern Europe"* contract I/76 887.

2 Results of the Project

In this section we present the main results from the project. We divided them into three groups: *language technology*, *language data*, and *software systems*. Some of the groups contain resources or tools that were made outside the project, but which were used within the project and were intensively validated and extended during their usage. Language technology concerns linguistic tools for processing of texts (written or transcribed from speech records). Language data is a set of corpora, grammars and lexicons which allow the usage of the language technology tools and their validation. Software systems are the software programs which support the implementation of the language technology tools, the creation and management of language data. Here we present our system for corpora development — CLaRK (see Section 2.3).

2.1 BulTreeBank Language Technology

Tokenization

We have implemented a hierarchy of tokenizers within the CLaRK system, which tokenize the texts in an appropriate way. Here we define the basic categories of tokens in Bulgarian texts. These tokens are the basis of the next steps of processing.

The Morphosyntactic analyzer

It assigns all possible morphosyntactic analyses to the word tokens. It is based on the morphological dictionary described below. From the morphological dictionary we generated all wordforms with their morphosyntactic characteristics. On the basis of this set of wordforms we have constructed a regular grammar in the CLaRK system. This regular grammar is used for the actual analysis. For some productive cases guessing rules have been implemented. In the places where competing analyses arised between a common word and a name or an abbreviation, we tried to use the token classification strategy and the prompts of the context. If there was no clear preference, we left the decision to the human annotator (in cases when the text would be manually checked). Otherwise we selected the most probable solution.

MorphoSyntactic Disambiguation

We have already implemented a preliminary version of a rule-based morpho-syntactic disambiguator, encoded as a set of constraints within the CLaRK system. This rule-based disambiguator exploits context information like *agreement between an adjective and a noun in a noun phrase*, specific positions like *a noun after a preposition*, but it also deals with some fixed phrases. The disambiguator does not try to solve unsure cases, but leaves them for further processing. Its coverage is about 80 %. For the purposes of the treebank we have manually disambiguated the rest 20 %. For automatic disambiguation we have developed a neural-network-based disambiguator (see [Simov and Osenova, 2001]). It achieves accuracy of 95.25% for part-of-speech and 93.17% for complete morpho-syntactic disambiguation.

Partial Grammars

We have constructed such grammars for:

1. **Sentence splitting.** At the moment it is fully automated and reliable only for the basic and clear cases. For solving complex and ambiguous cases this grammar is combined with supporting modules for abbreviation detection.
2. **Named-entity recognition.** Identifying numerical expressions, names, abbreviations, special symbols (see [Ivanova and Dojkoff 2002], [Osenova and Kolkovska 2002]). They are designed to work in cooperation with the morphosyntactic analyzer. If necessary, the grammars can overwrite the analysis of the morphosyntactic analyzer.
3. **Chunking.** Two basic modules have been developed: an Noun Phrase chunker (see [Osenova 2002], [Osenova and Kolkovska 2002]) and a Verb Phrase chunker ([Slavcheva 2002]). Generally speaking, the chunking process conforms to the following requirements: it deals with non-recursive

constituents; relies on a clear-indicator strategy; delays the attachment decisions; ignores the semantic information; aims at accuracy, not coverage. Additionally, there are chunk grammars for APs, AdvPs, PPs and some non-problematic clauses.

2.2 BulTreeBank Language Data

2.2.1 BulTreeBank Corpus

The text archive

It is intended to yield the size of a national corpus, that is, 100 million running words. Since the data are gradually annotated, its status at the moment is approximately as follows:

1. Nearly 90 million running words are collected from different sources in HTML and RTF formats. In order to compile a representative and balanced corpus of Bulgarian texts, we tried to gather a variety of different genres: 15% fiction, 78% newspapers and 7% legal texts, government bulletins and others.

2. About 72 million running words are converted into XML documents, marked up in conformance with the TEI guidelines. This conversion is automatic: for each source of text we developed a separate tool for extraction of the relevant information like the text itself, but also the author information, genre classification (where it is available), and other meta-information. The tools are implemented in Prolog and the CLaRK system.

3. 10 million running words are morphologically analyzed. This part of the text archive was used to select data for manual disambiguation and in future it will be substituted by an automatically disambiguated version of the full text archive.

4. Over 1 000 000 running words are morphosyntactically disambiguated by hand. This part of the text archive is used in two ways within the project: (1) as a source of sentences and articles which to be annotated syntactically and included in the treebank, and (2) as training and testing data for POS disambiguation of Bulgarian texts. The segmentation of the text and the morphosyntactic tagset used in this annotation are described in separate technical reports.

The Treebank

The Treebank (about 200 000 words) is a part of the BulTreeBank corpus. The annotation scheme of the treebank is described as a separate technical report. It is meant to be syntactically processed and consists of two layers:

1. Core set of sentences (1 500) - these are sentences, extracted mainly from Bulgarian grammars. They will serve as a test suite and gold standard for Bulgarian, because they are considered to represent the variety of the linguistic phenomena in our language.

2. Treebank (up to now 10 000) - these are sentences, extracted mainly from the electronic archive. First, they are pre-processed automatically, then the attachment operations are performed by the annotators. Note that the annotators are restricted by the software device and thus the analyses are consistent at this level. Finally, the sentences are post-edited and corrected.

2.2.2 Lexicons

The Morphological Dictionary

The dictionary is an electronic version of [Popov, Simov and Vidinska 1998] extended with new words from the corpus. It covers the grammatical information of about 100 000 lexemes (1 600 000 word forms) and serves as a basis for the morphological analyzer.

The Gazetteers

Two basic lists with items, missing in the morphological dictionary, have been compiled with respect to their frequency:

1. Gazetteers of names. These consist of 15 000 items and include Bulgarian as well as foreign person names, international and national locations, organizations. The most frequent names are additionally classified according to three criteria: (1) grammatical (gender and number); (2) semantic - with respect to an extended SIMPLE core ontology (names for different types of locations, organizations, artifacts, persons' social roles etc.) and (3) ontological - some person names were connected with specific individuals in the world and thus some encyclopedic information was provided in addition to the semantic classification. All this information can be used for practical applications like Information Extraction or Retrieval, Data Mining, etc. In the process of the construction of the treebank we envisage to use it for agreement specification and semantic selection. Special attention is paid to the names of mountains and artifacts (books, films, broadcasts), because their internal agreement does not always coincide with the external one, which is needed for the sentence analysis.

2. Gazetteers of the most frequent abbreviations. They consist of 1500 acronyms and graphical abbreviations. The acronyms' extensions were mapped against the names (mostly organizations) and therefore, assigned the same semantic and grammatical label. In cases of idiosyncratic grammatical behaviour, the relevant patterns have been added as well.

3. Gazetteers of the most frequent introductory expressions and parentheticals. This is considered to be a step towards a basic list of collocations. They were classified according to their morphological type or behavior: verbal, adverbial, linking (for conjunctions), nominal (vocatives), idiomatic etc. We use them as an extended supplementary lexicon during the phase of the syntactic annotation.

The Valence Dictionary

It consists of 1000 most frequent verbs and their valence frames and it is based on a paper dictionary (see [Balabanova and Ivanova, 2002]). Each frame defines the number and the kind of the arguments and imposes morphosyntactic and semantic restrictions over them. The semantic restrictions over the arguments are extracted and matched against the SIMPLE core ontology. The frames of the most frequent verbs are compared to the corpus data and repaired if necessary (new frames are added, some of the existing frames are deleted or fine-grained). We envisage to enlarge the coverage of the dictionary with the help of some derivational means, such as the verb prefixes.

The Semantic Dictionary

Semantic information plays a crucial role in the process of parse discrimination on which the construction of our treebank depends. Thus, in order to support the selectional restrictions imposed by the valence dictionary and to facilitate its usage, we decided to compile a semantic dictionary along the guidelines of SIMPLE project. It is worth mentioning that we follow an extended variant of the SIMPLE core ontology. At the moment we are classifying the most frequent nouns with respect to the ontological hierarchy without specifying the synonymic relations between them. Up to now we have classified about 3 000 nouns. Recall that the named entities also have been classified with respect to the same ontology.

2.3 CLaRK System

In this section we describe the basic technologies of the CLaRK System[1] ([Simov et. al. 2001]). CLaRK is an XML-based software system for corpora development. A complete description of the system is presented in a separate technical report. It incorporates several technologies: *XML technology*; *Unicode*; *Regular Grammars*; and *Constraints over XML Documents*.

XML Technology

[1] For the latest version of the system see `http://www.bultreebank.org/clark/index.html`.

The XML technology is at the heart of the CLaRK System. It is implemented as a set of utilities for data structuring, manipulation and management. We have chosen the XML technology because of its popularity, its ease of understanding and its already wide use in description of linguistic information. In addition to the XML language [XML 2000] processor itself, we have implemented an XPath language [XPath 1999] engine for navigation in documents and an XSLT engine [XSLT 1999] for transformation of XML documents. We started with basic facilities for creation, editing, storing and querying XML documents and developed further this inventory towards a powerful system for processing not only single XML documents but an integrated set of documents and constraints over them. The main goal of this development is to allow the user to add the desirable semantics to the XML documents. The XPath language is used extensively to direct the processing of the document pointing where to apply a certain tool. It is also used to check whether some conditions are present in a set of documents.

Tokenization

The CLaRK System supports a user-defined hierarchy of tokenizers. At the very basic level the user can define a tokenizer in terms of a set of token types. In this basic tokenizer each token type is defined by a set of UNICODE symbols. Above this basic level tokenizers the user can define other tokenizers for which the token types are defined as regular expressions over the tokens of some other tokenizer, the so called parent tokenizer. For each tokenizer an alphabetical order over the token types is defined. This order is used for operations like the comparison between two tokens, sorting and similar.

Regular Grammars

The regular grammars in CLaRK System [Simov, Kouylekov and Simov 2002] work over token and element values generated from the content of an XML document and they incorporate their results back in the document as XML mark-up. The tokens are determined by the corresponding tokenizer. The element values are defined with the help of XPath expressions, which determine the important information for each element. In the grammars, the token and element values are described by token and element descriptions. These descriptions could contain wildcard symbols and variables. The variables are shared among the token descriptions within a regular expression and can be used for the treatment of phenomena like agreement. The grammars are applied in cascaded manner. The evaluation of the regular expressions, which define the rules, can be guided by the user. We allow the following strategies for evaluation: 'longest match', 'shortest match' and several backtracking strategies.

Constraints over XML Documents

The constraints that we have implemented in the CLaRK System are generally based on the XPath language (see [Simov, Simov and Kouylekov 2003]). We use XPath expressions to determine some data within one or several XML documents and thus we evaluate some predicates over the data. Generally, there are two modes of using a constraint. In the first mode the constraint is used for validity check, similar to the validity check, which is based on a DTD or an XML schema. In the second mode, the constraint is used to support the change of the document to satisfy the constraint. The constraints in the CLaRK System are defined in the following way: `(Selector, Condition, Event, Action)`, where the selector defines to which node(s) in the document the constraint is applicable; the condition defines the state of the document when the constraint is applied. The condition is stated as an XPath expression, which is evaluated with respect to each node, selected by the selector. If the result from the evaluation is improved, then the constraint is applied; the event defines when this constraint is checked for application. Such events can be: selection of a menu item, pressing of key shortcut, an editing command; the action defines the way of the actual constraint application.

Cascaded Processing

The central idea behind the CLaRK System is that every XML document can be seen as a "blackboard" on which different tools write some information, reorder it or delete it. The user can arrange the applications of the different tools to achieve the required processing. This possibility is called **cascaded processing**. For more on application construction abilities of CLaRK System see [Simov, Simov and Osenova 2004].

3 Overview of the Reports

In this section we present the rest of the technical reports of the project.

Report *BTB-TR02: BulTreeBank Text Corpus of Bulgarian: Content, Segmentation, Tokenization* presents in detail the text corpus compiled within the project, its classification with respect to different text types, the structuring of the documents in the archive, and tokenization. Also, the sentence segmentation is considered because it plays an important role in the construction of the treebank.

Report *BTB-TR03: BulTreeBank Morphosyntactic Tagset* presents the tagset used in the morphosyntactic annotation of the treebank and the morphosyntactic corpus. The tagset follows the principles of the MULTEXT-EAST tagset for Bulgarian. It is positional and each position corresponds to a morphosyntactic category. For each tag there is an example wordform. The tagset consists of 659 tags.

Report *BTB-TR04: BulTreeBank Morphosyntactic Annotation of Bulgarian Texts* contains guidelines for the morphosyntactic annotation of the various kinds of tokens. Also a list of the most frequent ambiguities at this level is given. For each ambiguity rules for their disambiguation are presented.

Report *BTB-TR05: BulTreeBank Style Book* contains the guidelines for the syntactic annotation in the treebank. First, it presents the HPSG language model as a basis for the annotation scheme. Then the syntactic domains (Noun Phrases, Verb Phrases, etc) are presented together with the head-dependant relations within each domain. The next part lists the language phenomena that are explicated within the treebank. The report concludes with a list of preference rules for some of the ambiguous cases.

Report *BTB-TR06: CLaRK — an XML-based System for Corpora Development* is a general overview of the CLaRK System. First, it presents the basic technologies behind the system. Then a description of each tool of the system follows. Additionally to the report a user manual can be found on the web page of the system: `http://www.bultreebank.org/clark/index.html`

References

[Balabanova and Ivanova, 2002] Elisaveta Balabanova and Krassimira Ivanova. 2002. *Creating a machine-readable version of Bulgarian valence dictionary: (A case study of CLaRK system application).* In: *Proc. of The First Workshop on Treebanks and Linguistic Theories.* Sozopol, Bulgaria.

[Ivanova and Dojkoff 2002] Krassimira Ivanova and Dimitar Doikoff. 2002. *Cascaded Regular Grammars and Constraints over Morphologically Annotated Data for Ambiguity Resolution.* In: *Proc. of The Workshop on Treebanks and Linguistic Theories.* Sozopol, Bulgaria.

[Osenova 2002] Petya Osenova. 2002. *Bulgarian Nominal Chunks and Mapping Strategies for Deeper Syntactic Analyses.* In: *Proc. of The Workshop on Treebanks and Linguistic Theories.* Sozopol, Bulgaria.

[Osenova and Kolkovska 2002] Petya Osenova and Sia Kolkovska. 2002. *Combining the named-entity recognition task and NP chunking strategy for robust pre-processing.* In: *Proc. of The Workshop on Treebanks and Linguistic Theories.* Sozopol, Bulgaria.

[Osenova and Simov 2002] Petya Osenova and Kiril Simov. 2002. *Learning a token classification from a large corpus. (A case study in abbreviations).* In: *Proc. of the ESSLLI Workshop on Machine Learning Approaches in Computational Linguistics*, Trento, Italy.

[Popov, Simov and Vidinska 1998] Dimitar Popov, Kiril Simov and Svetlomira Vidinska. 1998. *A Dictionary of Writing, Pronunciation and Punctuation of Bulgarian Language.* Atlantis LK, Sofia, Bulgaria.

[Simov et. al. 2001] Kiril Simov, Zdravko Peev, Milen Kouylekov, Alexander Simov, Marin Dimitrov, Atanas Kiryakov. 2001. *CLaRK - an XML-based System for Corpora Development.* In: Proc. of the Corpus Linguistics 2001 Conference. pp 558–560.

[Simov and Osenova, 2001] Kiril Simov and Petya Osenova. 2001. *A Hybrid System for MorphoSyntactic Disambiguation in Bulgarian.* In: Proc. of the RANLP 2001, Tzigov chark, Bulgaria.

[Simov, Kouylekov and Simov 2002] Kiril Simov, Milen Kouylekov, Alexander Simov. 2002. *Cascaded Regular Grammars over XML Documents.* In: Proc. of the 2nd Workshop on NLP and XML (NLPXML-2002), Taipei, Taiwan.

[Simov, Simov and Kouylekov 2003] Kiril Simov, Alexander Simov, Milen Kouylekov. 2003. *Constraints for Corpora Development and Validation.* In: Proc. of the Corpus Linguistics 2003 Conference, pages: 698–705.

[Simov, Simov and Osenova 2004] Kiril Simov, Alexander Simov, Petya Osenova. 2004. *An XML Architecture for Shallow and Deep Processing.* In: Proc. of the ESSLLI 2004 Workshop on Combining Shallow and Deep Processing for NLP. Nancy, France. pages: 51–60.

[Slavcheva 2002] Milena Slavcheva. 2002. *Segmentation Layers in the Group of the Predicate: a Case Study of Bulgarian within the BulTreeBank Framework.* In: *Proc. of The Workshop on Treebanks and Linguistic Theories.* Sozopol, Bulgaria.

[XML 2000] XML. 2000. *Extensible Markup Language (XML) 1.0 (Second Edition).* W3C Recommendation. http://www.w3.org/TR/REC-xml

[XPath 1999] XPath. 1999. *XML Path Language (XPath) version 1.0.* W3C Recommendation. http://www.w3.org/TR/xpath

[XSLT 1999] XSLT. 1999. *XSL Transformations (XSLT). version 1.0.* W3C Recommendation. http://www.w3.org/TR/xslt