BTB-TR02: BulTreeBank Text Corpus of Bulgarian: Content, Segmentation, Tokenization*

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

BulTreeBank Technical Report BTB-TR02

05.04.2004

Abstract

This document contains a description of the BulTreeBank corpus and its processing. The corpus consists of a text archive, morphologically annotated corpus and a treebank. In this report we outline the general structure of the corpus as well as the segmentation of the texts in sentences and tokenization. In other technical reports we describe the annotation at morphological and syntactic level.

1 Introduction

We aimed at the creation of a linguistically interpreted corpus for Bulgarian. It means that every token would receive some linguistic interpretation at some or all language levels: token type (common word, abbreviation, name, symbol), part-of-speech tag and morphological features, syntactic behavior. For that purpose an appropriate structural representation was needed first. We chose XML mark-up language and TEI specifications (see [TEI 1997]) for the layout of the text structure.

When we started the project we had at our disposal a text archive of Bulgarian texts collected from the Internet. These texts covered about 10 000 000 running words. The texts were converted into TEI compatible XML markup at the paragraph level. 16% of the texts came from fiction, 81% from newspapers and about 3% from other genres. We considered this initial text archive too small for some of the tasks we wanted to pursue in the project. Especially for statistical extraction of abbreviations, names, collocations and similar text elements. Thus one of the first tasks within the project was to extend the text archive.

The BulTreeank Text Archive (BTB-TA) 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 were 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

*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.

variety of different genres: 15% fiction, 78% newspapers and 7% legal texts, government bulletins and others.

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

3. 10 million running words were 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 were morphosyntactically disambiguated by hand. This part of the text archive has been 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.

In the text archive each document has to be marked-up at least to the structural level: chapters, articles, paragraphs. The sub-paragraph level of annotation is performed just on part of the corpus. This level includes (1) the tokenization of the text into tokens which are potential wordforms and punctuation, (2) segmentation of the text into sentences; and (3) the syntactic annotation itself.

Here we present the first two levels because they are common for the morphosyntactic corpus and the treebank itself. Usually the first two levels are considered as one task because they are interconnected, but for consistency we separate them here.

2 Tokenization

As it was mentioned above, the tokenization is the process of segmentation of the text into sentences and words: [Grefenstette and Tapanainen 1994]. In general, the task is quite complex and in CLaRK System we divided it between two tools: tokenizers and regular grammars. The tokenizers work in a cascaded manner. First, a primitive tokenizer is executed which assigns a category to each Unicode symbol, then a sequence of complex tokenizers is applied. Each tokenizer in the sequence works over the output of the previous tokenizer. The tokens for each complex tokenizer are defined via regular expressions. The result of the tokenization is a list of tokens and their categories. This list is an input to the regular grammar tool which actually annotates the text if necessary. At the tokenization level our goal is to segment the text into a list of potential words, punctuation, numerical expressions. We assume that abbreviations, sentences, dates and similar entities are processed at the next level although one can try to do this directly at the tokenization level. Some of the trickier cases like several words forming one multi-tokens also can be processed later.

We first tokenized the texts with respect to the following categories:

1. `CyrWord`. Cyrillic tokens consist of one or more Cyrillic letters with optional internal dashes or apostrophe. Here are some examples: *люляк*, *министър-председател*, *ВиК*, *МнВР*, *Весела*, *О'Хенри*, *к'во*.

2. `CyrWord-`. Cyrillic tokens of one or more Cyrillic letters with optional internal dashes and obligatory end dash. Here are some examples: *радио-*, *водо-*. These are separated from the `CyrWord` tokens because they have a separate treatment.

3. `LatWord`. Latin tokens consist of one or more Latin letters with optional internal dashes or apostrophe. Here are some examples: *lilac*, *light-hearted*, *GmbH*, *Brown*.

4. `LatWord-`. Latin tokens of one or more Latin letters with optional internal dashes and obligatory end dash. These are separated from the `LatWord` tokens because they have a separate treatment.

5. `Num`. Numerical tokens consist of one or more digits with optional internal dots, commas, spaces, and dashes, leading minus and plus sign. Here are some examples: *123*, *3.14159*, *1991-1998*, *21.03.2000*, *195.96.243.78*, *72 000 000*, *+395 2 979-28-25*.

6. `AlphaNum`. Tokens consist of Cyrillic letters, Latin letters, digits, dashes. Here are some examples: *МиГ-29*, *US-журналисти*, *9-годишен*, *Атина'04*.

7. `Punct`. Any sequences of punctuation marks. Here we include dot `.` (when it is not part of other tokens), comma `,` (when it is not part of other tokens), question mark `?`, exclamation mark `!`, colon `:`, semicolon `:`, brackets `( ) [ ] { }`, dash `-` (when it is not part of other tokens).

8. `Internet`. Here we include all URL and e-mail addresses. For example: *neworg.prg/~user*, *http://www.bultreebank.org*, *kivs@bultreebank.org*.

9. `Space`. Any single space, tabulation or new line symbols.

10. `Symbol`. Any single symbol which is not covered by the above categories.

Additionally, we classified each kind of token into general token categories like common words, proper names, abbreviations and punctuation. Special attention we paid to the subclassification of `CyrWord` class of tokens because it contains the majority of the ambiguous token classes. For an initial token classification see [Osenova and Simov 2002]. On the basis of this initial classification we created gazetteers for proper names and abbreviations. Similarly we classified the tokens from the other classes with respect to these categories. The classification was made on basis of lexicons, gazetteers and regular grammars. Thus for `CyrWord-` we created a lexicon of word prefixes, because they are used in special kind of coordination expressions like: *радио- и телевизионни програми*, *водо- и топлоцентрали*. `LatWord` and `LatWord-` were divided into two groups: (1) well-known Latin words and phrases (*Nato*, *MS Windows*, *USD*, *CHF*); (2) others. For the first group we prepared lexicons where they are treated as common words, proper names or abbreviations. The tokens of the second group were left as they are in the text, but see also [Simov and Osenova 2004]. `Num` tokens are treated as common words or names (telephone numbers, dates). `AlphaNum` class also contains token from different categories: common words — *9-годишен*, abbreviations — *МиГ-29*, proper names — *Атина'04*. The tokens of class `Internet` are considered as proper names. Tokens of class `Symbol` can be common words (*%*, *$*) or punctuation marks (*/*). How the corresponding categories of tokens are annotated in the morphosyntactic corpus see in [Simov and Osenova 2004].

3 Sentence Segmentation

At first sight the sentence segmentation seems to be a simple problem. The naive approach is that the sentence equals each sequence of tokens starting with a capitalized common word, proper name or abbreviation and ending in a full stop, question mark, exclamation mark. This means that punctuation marks are divided into two groups: (1) delimitating the end of a sentence and (2) internal to a sentence. However, these groups can intersect. For example, period, exclamation mark, question mark etc. can be used inside the sentence. Sometimes, sentences in the text do not even end in a punctuation mark. Finally, some sentences in the text can be considered as elements of another sentence. Thus, in practice sentence delimitation turns out to be a much more complicated task. Here we present some examples of such problematic cases and how we solved them in the morphosyntactic corpus and the treebank.

When the text are tokenized we keep all the punctuation because it plays an important role in the determination of the morphosyntactic features and syntactic structure of the words and the sentences. Thus it is accepted that question mark, exclamation mark, and period can be also internal punctuation marks. For instance, the following sentence

```
<s>Ще взема един камък и бух! в прозореца.</s>
```

is tokenized as:

```
<s>
    <tok>Ще</tok>
    <tok>взема</tok>
    <tok>един</tok>
    <tok>камък</tok>
    <tok>и</tok>
    <tok>бух</tok>
    <pt>!</pt>
    <tok>в</tok>
    <tok>прозореца</tok>
    <pt>.</pt>
</s>
```

Here the token categories are not shown and the punctuation is annotated as `<pt>` element.

The main problem in this respect is the period when it is also a part of an abbreviation. Then we would like to keep the period as a part of the abbreviation. Thus, for example, the sentence

```
<s>Те разчитали на факта, че от 1997 г. ходжите не
   са получавали заплати.</s>
```

contains the abbreviation `г.` which is recognized as one token:

```
<s>
    <w>Те</w>
    <w>разчитали</w>
    <w>на</w>
    <w>факта</w>
    <pt>,</pt>
    <w>че</w>
    <w>от</w>
    <w>1997</w>
    <abbr>г.</abbr>
    <w>ходжите</w>
    <w>не</w>
    <w>са</w>
    <w>получавали</w>
    <w>заплати</w>
    <pt>.</pt>
</s>
```

Here the tokens are classified as common words (`<w>`), abbreviations (`abbr`) and punctuation (`<pt>`).

The same abbreviation at the end of a sentence will be segmented into two tokens: the abbreviation and a full stop. Here is an example. The sentence

```
<s>Бил в Тимишоара с бригада хирурзи през 1989 г.</s>
```

is tokenized as follows:

```
<s>
    <w>Бил</w>
    <w>в</w>
    <name>Тимишоара</name>
    <w>с</w>
    <w>бригада</w>
    <w>хирурзи</w>
    <w>през</w>
    <w>1989</w>
    <abbr>г.</abbr>
    <pt>.</pt>
</s>
```

Note that here the full stop is presented twice — as a part of the abbreviation and as a full stop for the sentence.

One of the hard nuts for sentence splitting task is the representation of direct speech and authors' introductory words. Very often the author's introductory words separate in different way the direct speech. In all the cases the direct speech sentences appear to be complements of the introductory part. Thus, from one point of view it is preferable to encode such sentences as parts of one larger sentence. On the other hand, this step could create very extensive and incomprehensible structures. Thus, we decided to use a mixed approach: *If the dependent sentence is rather short, then we do not separate this sentence.* However, if the sentence is too long or there are several sentences in chain, then we separate all of them including the speech of the author. Here we give several examples with increasing complexity.

In the following example, the dependent sentence is not segmented as a sentence. Such segmentation will be done when the syntactic structure is annotated:

```
<s>Народът казва "Рибата се вмирисва откъм главата".</s>
```

Later on, the dependent sentence is annotated as a clause:

```
<s>Народът казва <CL>"Рибата се вмирисва откъм главата"</CL>.</s>
```

When the dependent sentence is too long or there are more than one dependent sentences, we segment each sentence including the main one. Here are some examples:

The paragraph:

```
<p>"Искаме България да стане първата страна от
    Югоизточна Европа, която да издава и трите
    треньорски степени", заяви той.</p>
```

is segmented in the following way:

```
<p><s>"Искаме България да стане първата страна от
       Югоизточна Европа, която да издава и трите
       треньорски степени"</s>
   <s>, заяви той.</s></p>
```

Here the surrounding quotation marks are captured within the dependent sentence. When it is not possible we leave them outside the `<s>` elements. Here is an example of such a paragraph:

```
<p>"В 15-годишната си кариера Сотомайор е проверяван
```

```
   над 300 пъти и всички проби са били отрицателни.
   Взехме предвид и факта, че участва в много
   благотворителни акции и че това ще е последната
   му олимпиада", обяви говорителят на ИААФ Джорджо
   Рейнери.</p>

<p>
    "
    <s>В 15-годишната си кариера Сотомайор е проверяван
       над 300 пъти и всички проби са били отрицателни.</s>
    <s>Взехме предвид и факта, че участва в много
       благотворителни акции и че това ще е последната
       му олимпиада</s>
    "
    <s>, обяви говорителят на ИААФ Джорджо Рейнери.</s>
</p>
```

When the author's speech separates the direct speech we again segment each sentence in a single <s> element, but more punctuation can be presented outside the sentences. Thus some sentences can loose their own full stops:

```
<p>"Хавиер е много щастлив
   - каза съпругата му Мария дел Кармен Гарсия. -
   Правосъдието възтържествува".</p>

<p>
    "
    <s>Хавиер е много щастлив</>
    <s>- каза съпругата му Мария дел Кармен Гарсия. -</s>
    <s>Правосъдието възтържествува</s>
    ".
</p>
```

Later on, at the syntactic level the dependent sentences are analysed as complements of the main one.

4 Conclusion

We have created a multi-layer corpus of linguistically interpreted texts. The collected database reflects various sources, such as: newspapers (national and local ones), prose (original and translated), government documents, miscellaneous.

The creation of such a corpus required an adequate pre-processing at tokenization and segmentation level.

References

[Grefenstette and Tapanainen 1994] Gregory Grefenstette and Pasi Tapanainen. 1994. *What is a word, What is a sentence? Problems of Tokenization.* In: Proc. of The 3rd International Conference on Computational Lexicography (COMPLEX'94). Budapest, Hungary. pp 79–87.

[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.

[Simov and Osenova 2004] Kiril Simov and Petya Osenova. 2004. *BTB-TR04: BulTreeBank Morphosyntactic Annotation of Bulgarian Texts* BulTreeBank Technical Report BTB-TR04.

[Simov et. al. 2004] Kiril Simov, Petya Osenova, Milena Slavcheva. 2004. *BTB-TR03: BulTreeBank Morphosyntactic Tagset. BTB-TS version 2.0.* BulTreeBank Technical Report BTB-TR03.

[TEI 1997] Text Encoding Initiative. 1997. Guidelines for Electronic Text Encoding and Interchange. Edited by C. M. Sperberg-McQueen and Lou Burnard. `http://www.tei-c.org/uic/ftp/P4beta/index.htm`