Bulgarian Pipeline — BTBPipe

Kiril Simov, Petya Osenova, Iliana Simova

Linguistic Modelling Laboratory, IICT-BAS, Sofia, Bulgaria
{kivs|petya|iliana}@bultreebank.org

The Bulgarian language pipeline is developed as part of several European projects[1] using language resources created within BulTreeBank group at the Language Modelling Division of the Institute of Information and Communication Technology at Bulgarian Academy of Sciences.

The current pipeline includes the following modules:

1 Tokenizer and Sentence Splitter

The input to the module is in plain text format. The module is implemented as a cascaded regular grammar and a set of rules in CLaRK System[2]. The start and end positions of each token are stored.

2 Part-of-Speech Tagger

The POS tagging in Bulgarian is more complex than the same task in English. Bulgarian is also an analytical language, but with rich word inflection. Although we often refer to this task as POS tagging, it is more accurate for it to be defined as morphosyntactic annotation or morphological tagging, because of the big variety of grammatical features and their interdependance. To tackle the complexity of the problem in an adequate way we use the full form of 680 tags of the BulTreeBank Morphosyntactic Tagset (BTB-TS)[3], which is the original tagset of the BulTreeBank.

2.1 Rule-based Module for Morphological Tagging

The rule-base module for morphological tagging exploits two sources of linguistic knowledge: the morphological lexicon (near 110000 lemmas) and the

[1] The recent projects are: EuroMatrixPlus, EUCases and QTLeap projects.

[2] http://www.bultreebank.org/clark/index.html

[3] http://www.bultreebank.org/TechRep/BTB-TR03.pdf

gazetteers (more than 26000 names), and 70 disambiguation rules, implemented in the CLaRK System. The rules are hand-crafted and then arranged as an algorithm in a specific order. The rules are extensively tested during the creation of BulTreeBank and they produce 100% accuracy result.

These rules work on an input in which the tokens are annotated with all possible tags provided by the morphological lexicon. First the algorithm looks up the morphological dictionary and retrieves all possible tags for each token in the text. Then the rules can narrow down the possible tags for a given word by selecting one of the possible tags. In the rest of the cases all possible tags remain in the annotation. They were designed to achieve higher precision even at the cost of low recall.

Here is an example of a rule: If a wordform is ambiguous between a masculine count noun (Ncmt) and a singular short definite masculine noun (Ncmsh), the Ncmt tag should be chosen if the previous token is a numeral or a number.

The result of this component is represented in table format:

```
w1  ListOfTags1
w2  ListOfTags2
...
wn  ListOfTagsn
```

Where ListOfTags is one or more morphosyntactic tags separated by semicolon. These tags are taken from the morphological lexicon. For unknown words we guess the most appropriate tags. Then in some cases the rules reduce the number of ambiguities. These tokens are used by the following statistical tagger.

2.2 Guided Learning System — GTagger

We used the guided learning framework described in [Shen et al. 2007], which has yielded state-of-the-art results for English and has been successfully applied to other morphologically complex languages such as Icelandic [Dredze and Wallenberg 2008]; we found it quite suitable for Bulgarian as well. We used the feature set defined in [Shen et al. 2007], which includes the following:

1. The feature set of [Ratnaparkhi 1996], including prefix, suffix and lexical, as well as some bigram and trigram context features;

2. Feature templates as in [Ratnaparkhi 1996], which have been shown helpful in bidirectional search;

3. More bigram and trigram features and bi-lexical features as in [Shen et al. 2007].

The tagger is using the input list of tags as features. They are used in two modes: soft and hard constraints. In the first mode they are used as additional features, but the suggested tag could come outside of the list. In hard constrains mode the suggested tag has to be among these on the list.

Thus, the GTagger is working on the output of rule-based module. When there is only one tag on the input it is copied to the output. In case of ambiguity the GTagger select one of the input tags. The best accuracy results was achieved by the system using hard constraints mode: 97.98%.

3 Lemmatizer

The lemmatization module comprises a set of transformation rules that we have developed, based on the morphological lexicon (see Figure 1). They were implemented via finite state automata in the CLaRK system instead of word forms directly being looked up in the lexicon. We motivate our decision with its faster operation speed. Furthermore, the rules were based on the morphological dictionary, presented above. We also believe that these rules can be used on unknown words in order to produce some guessing about their word lemmas.

In theory, the lemmatization should be a deterministic process, but in some cases more than one lemma is assigned to a word form. This outcome can be expected when the word form is ambiguous with respect to its base form, and the disambiguation process requires some bigger context or other type of analysis. In these cases the lemmatizer will let the decision to be postponed for a later stage of analysis.

a. *if* **pos-tag** = **POS-Tag** *then*
{*remove* **OldSuffix**; *concatenate* **NewSuffix**}

b. *if* **pos-tag** = **Vpitf-o1s** *then*
{*remove* **-oh**;*concatenate* **-a**}

Figure 1: Examples of lemmatization transformation rules in a. and replacement rule for *chetoh* (I read/Past) in b.

Combining the lemmatization rules with the best result on morphological tagging results in more than 95 % accuracy.

4 Mate tools parser

For statistical dependency parsing we are using MATE tools parser. This choice is motivated by the results from an experiment with several dependency parsing tools on Bulgarian data, in which Mate tools achieved the best performance. The Unlabeled Accuracy Score for dependency parser is 92.9 %. MATE tools parser was trained on BulTreeBank data[4].

5 NERC and NED modules

This and the following modules are new for the Bulgarian pipeline. This means that we expect to have better implementation for all of them in 2015. Also we will be working on the extension of the coverage of the language datasets used in implementation of the modules.

The Bulgarian Named Entity Recognition and Classification (NERC) is a rule-based module. It uses a gazetteer with names categorized in four types: Person, Location, Organization, Other. The identification of new names is based on two factors - sure positions in the text and classifying contextual information, such as, titles for persons, types of geographical objects or organizations, etc. The disambiguation module uses simple unigram-based statistics. Additionally, in Named Entity Disambiguation (NED) module part of the names are connected to Bulgarian DBPedia instance URIs and classes. DBpedia's ontological hierarchy determines the more general categories for DBpedia instances (City, Politician, etc.) as subclasses of Person, Location and Organization. For other kinds of instances we rely on the most general category provided by the classification of the instance according to the DBpedia ontology. The unigram-based statistics is adapted to the new categories. In case the selected categories in the annotation are not sufficient for disambiguating among DBpedia instance URIs, we store all of them in the annotation.

For the evaluation of the NERC module we manually checked the performance on new text (12223 tokens). The gold standard annotation contains 810 named entities. The automatic procedure recognized 688 entities, the intersection annotations with the gold standard were 593. The precision of the tool is 86.1 % and the recall is 73.2 %. During the rest of the project we will be improving the tool by adding more names to the gazetteers in use and by creating better rules for multiword names.

For the NED module we have reused the same data for measurement as in the case of NERC. The gold standard annotations of DBpedia instances are 667. The automatic procedure annotated 391 instances. The intersection of the annotated instances is 248. Thus the precision of the tool is 63.43 % and the recall is 37.18 %. The low results are due to the small coverage of the Bulgarian DBpedia. In order to solve this problem in the next phase of the project we plan to extend the coverage of the Bulgarian DBpedia in the following ways: (1) using Bulgarian Wikipedia articles that are not in the Bulgarian DBpedia but have linked corresponding instances in the English DBpedia. In this case we will automatically transfer the ontological

[4] http://www.bultreebank.org/dpbtb/

classification from the English DBpedia to the new Bulgarian instances; (2) using transliteration rules, we will transliterate English instance names into Bulgarian ones. In the first case we will be able to refer to both Bulgarian and English Wikipedia articles. In the second case we will be able to refer only to English ones. The second approach could possibly introduce errors due to cases of wrong transliteration or ambiguous Bulgarian names.

It is an unfortunate fact that DBpedia Spotlight[5] does not support Bulgarian.

6 Word Sense Disambiguation

The basic version of WSD is implemented on the assumption of one sense per discourse and bigram statistics. In the next phase of the project more advanced system will be implemented using additional semantic resources like ontologies, base concepts of WordNet as well as syntactic structure of the sentences.

Again, we have reused the same data for measurement as in the case of NERC. The gold standard sense annotations are 3118. The automatic procedure annotated 2727 cases. The annotations in common are 1925. Precision is 70.6 % and recall is 61.7 %. The result is relatively good, bearing in mind the limited size of the Bulgarian WordNet, which was used in the annotation. We plan on improving the result by extending the coverage of the WordNet and by exploiting a better tool for WSD.

7 Coreference

We have implemented a basic version of a coreference resolution module, using paths in the dependency tree of each sentence. By using path patterns, we are mainly performing anaphora resolution. When dealing with the rest of the word forms, we consider the open class words that belong to the same synsets in WordNet and we group them together.

The same corpus was used for evaluating the module for coreference resolution. The human-annotated coreference chains are 337. The automatic annotation yielded 53 chains, a difference that is too large. One problem was identified as the source of this apparent bad performance: the automatic procedure selected coreference chains that were too long, because they extended beyond the boundaries of individual texts. In order to overcome this problem, we constructed lists of the coreferent words in each chain and used those to calculate performance. The gold standard annotations contain 903 related words. The automatic procedure returns 563 related words, out of which 371 match the gold standard data. Measured in this way, precision is

[5] https://github.com/dbpedia-spotlight/dbpedia-spotlight/wiki

65.9 and recall is 41.1 %. We hope we can achieve better results by exploiting other systems for solving the task.

8 Usages of the pipeline

Beside the IICT-BAS projects mentioned above BTB-Pipe is provided to projects and people outside IICT-BAS. The personal usage is usually by PhD students working on Bulgarian. The main projects that are using the pipeline are the following:

1. European project Pheme. Within the Pheme project the pipeline is used for processing of Bulgarian Tweets.

2. European project EXCITEMENT. Within the EXCITEMENT project the pipeline is used as part of a system for text entailment.

3. We also support the following activities on monitoring political speech with our tools for processing Bulgarian:

   - Organization: Open Knowledge Foundation - Bulgarian group

     Project: Automated topic and named entity extraction based on the based on the work and data collected in the Civil monitoring fair elections 2013 project of Institute for Social Integration - Sofia, in cooperation with FEPS: Foundation for European Progressive Studies - Brussels.
   - Organization: Open Knowledge Foundation - Bulgarian group and Ontotext AD

     Project: New Year's Speech of the President

     Description: Topic extraction from the previous presidential speeches and prediction about the topics and phrases which are to be used in the coming speech
   - Organization: Open Knowledge Foundation - Bulgarian group and Ontotext AD

     Project: Boyko Borisov Speech

     Description: Language analysis of all the former prime minister Boyko Borisov interviews and media appearances

     Publication: Interview for the National Radio and several daily newspapers publications Organization: Open Knowledge Foundation - Bulgarian group

     Project: "Pre-election day agitation press monitoring"

     Description: Monitoring and measuring the publications which may be considered political agitation in the pre-election day of 2013 parliamentary elections

   Publication: The results have been used in the claim brought before The Bulgarian constitutional court in Constitutional case No 13 from 2013.

4. Bulgarian project SINUS. Within the project the pipeline is used for processing descriptions of an icon collection.

5. Bulgarian project Culture of Charity in Education. The pipeline is used in a system for key terms extraction.

The pipeline exports the annotation results in NAF format[6].

References

[Dredze and Wallenberg 2008] Mark Dredze and Joel Wallenberg. 2008. Icelandic data driven part of speech tagging. In *Proceedings of the 44th Annual Meeting of the Association of Computational Linguistics: Short Papers*, ACL '08, pages 33–36, Columbus, Ohio, USA.

[Ratnaparkhi 1996] Adwait Ratnaparkhi. 1996. A maximum entropy model for part-of-speech tagging. In Eva Ejerhed and Ido Dagan, editors, *Fourth Workshop on Very Large Corpora*, pages 133–142, Copenhagen, Denmark.

[Shen et al. 2007] Libin Shen, Giorgio Satta, and Aravind Joshi. 2007. Guided learning for bidirectional sequence classification. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, ACL '07, pages 760–767, Prague, Czech Republic.

[6] http://wordpress.let.vupr.nl/naf/