Cross-lingual Adaptation as a Baseline: Adapting Maximum Entropy Models to Bulgarian

Georgi Georgiev
Ontotext AD
135 Tsarigradsko Chaussee
1784, Sofia
Bulgaria
georgi.georgiev@ontotext.com

Preslav Nakov
Department of Computer Science
National University of Singapore
13 Computing Drive
Singapore 117417
nakov@comp.nus.edu.sg

Petya Osenova and Kiril Simov
Linguistic Modelling Laboratory
Institute for Parallel Processing
Bulgarian Academy of Sciences
25A Acad. G. Bonchev St., 1113 Sofia, Bulgaria
{*petya,kivs*}@*bultreebank.org*

Abstract

We describe our efforts in adapting five basic natural language processing components to Bulgarian: sentence splitter, tokenizer, part-of-speech tagger, chunker, and syntactic parser. The components were originally developed for English within OpenNLP, an open source maximum entropy based machine learning toolkit, and were retrained based on manually annotated training data from the BulTreeBank. The evaluation results show an F_1 score of 92.54% for the sentence splitter, 98.49% for the tokenizer, 94.43% for the part-of-speech tagger, 84.60% for the chunker, and 77.56% for the syntactic parser, which should be interpreted as baseline for Bulgarian.

Keywords

Part-of-speech tagging, syntactic parsing, shallow parsing, chunking, tokenization, sentence splitting, maximum entropy.

1 Introduction

Nowadays, the dominant approach in natural language processing (NLP) is to acquire linguistic knowledge using machine learning methods. Other approaches, e.g., using manual rules, have proven to be both time-consuming and error-prone. Still, using machine learning has one major limitation: it requires manually annotated corpora as training data, which can be quite costly to create. Fortunately, for Bulgarian such a rich resource already exists – the BulTreeBank[1], an HPSG-based Syntactic Treebank with rich annotations at various linguistic levels. The existence of such a resource makes it possible to adapt to Bulgarian various NLP tools that have been originally developed for other languages, e.g., English, and that have been trained on similar kinds of resources, e.g., the Penn Treebank [4].

In this paper, we further stipulate that language adaption should be no harder than domain adaptation [2]. Similarly to Buyko & al. [2], we experiment with the OpenNLP tools[2] since they are open source and contain several platform-independent Java implementations of important NLP components. Moreover, these tools are based on a single machine learning algorithm, maximum entropy (ME) [1], as implemented in the OpenNLP MaxEnt[3] Java package. In our experiments below, we focus on five basic components from the OpenNLP tools: sentence detection, tokenization, part-of-speech (POS) tagging, chunking, and parsing.

Maximum entropy models search for a distribution $p(x|y)$ that is consistent with the empirical observations about a particular feature $f(x, y)$, computed from a set of training examples $\mathcal{T} = \{x, y\}$, e.g., a sentence x and its labeling y; see [6] for details. From all such distributions, the one with the highest entropy is chosen [1]. It can be shown that the resulting distribution will have the following form:

$$p_w(y|x) \propto \exp(w \cdot f(x, y)) \qquad (1)$$

The features used in the OpenNLP framework combine heterogeneous contextual information such as words around the end of a sentence for the English sentence splitter, or word, character n-grams and part-of-speech tag alone and in various combinations for the English chunker. These features are based on the publications of Sha and Pereira [7] for the chunker, and on the dissertation of Ratnaparkhi [6] for the POS tagger and the syntactic parser.

The remainder of this paper is organized as follows: Section 2 describes the process of converting the BulTreeBank XML data to Penn Treebank-style bracketing, Section 3 describes the experiments and discusses the results, and Section 4 concludes and suggests directions for future work.

[1] Created at the Linguistic Modelling Laboratory (LML), Institute for Parallel Processing, Bulgarian Academy of Sciences. See `http://www.bultreebank.org` for details.

[2] `http://opennlp.sourceforge.net`

[3] `http://maxent.sourceforge.net`

2 Converting the BulTreeBank XML to Penn Treebank-style bracketing

Converting the BulTreeBank XML [9, 5] to Penn Treebank-style bracketing format is straightforward, with some exceptions, for which we define custom tools. Consider, for example, the following sentence:

Всички на някакъв етап от живота си сме
изправени пред проблеми и предизвикателства .

```
‘All at some point of life itself we face
 to problems and challenges.’
```

```
We all at some point of our lives face
problems and challenges.
```

It has the following XML structure in BulTreeBank:

```
<S>
 <VPA>
  <VPS>
   <Pron>
    <w ana="Pce-op">Всички</w>
   </Pron>
   <PP>
    <Prep>
     <w ana="R">на</w>
    </Prep>
   </PP>
   <NPA>
    <NPA>
     <A>
      <w ana="Pfa--s-m">някакъв</w>
     </A>
     <N>
      <w ana="Ncmsi">етап</w>
     </N>
    </NPA>
    <PP>
     <Prep>
      <w ana="R">от</w>
     </Prep>
     <N>
      <w ana="Ncmsh">живота</w>
     </N>
     <Pron idref="id1">
      <w ana="Psxto">си</w>
     </Pron>
    </PP>
   </NPA>
   </PP>
   <VPC>
    <V>
     <w ana="Vxitf-r1p">сме</w>
    </V>
    <Participle>
     <w ana="Vpptcv--p-i">изправени</w>
    </Participle>
    <PP>
     <Prep>
      <w ana="R">пред</w>
     </Prep>
     <CoordP>
      <ConjArg>
       <N>
        <w ana="Ncmpi">проблеми</w>
       </N>
      </ConjArg>
      <Conj>
       <C>
        <w ana="Cp">и</w>
       </C>
      </Conj>
      <ConjArg>
       <N>
        <w ana="Ncnpi">предизвикателства</w>
       </N>
      </ConjArg>
     </CoordP>
    </PP>
   </VPC>
  </VPS>
 </VPA>
 <pt>.</pt>
</S>
```

We transform the above XML into the following Penn Treebank-style bracketing structure:

```
((S
 (NP
  (PRP Всички)
 )
 (VP
  (PP
   (IN на)
   (NP
    (NP
     (PRP някакъв)
     (NN етап)
    )
    (PP
     (IN от)
     (NN живота)
     (PP$ си)
    )
   )
  )
  (VP
   (VB сме)
   (VB изправени)
   (PP
    (IN пред)
    (NP
     (NP
      (NNS проблеми)
     )
     (CC и)
     (NP
      (NNS предизвикателства)
     )
    )
   )
  )
 )
 (. .)
```

In the process of transformation, we further apply some simple rules for the coordinations in the BulTreeBank, e.g., "CoordP" and "ConjArg" typically become "NP", and "Conj" becomes a "CC" phrase; see [5] for further details on the syntactic phrase naming conventions of the BulTreeBank and [4] for those of the Penn Treebank. We also remove the outer verb phrases in the BulTreeBank, e.g., the phrases "VPA" and "VPS" in the above example, as required by the Penn Treebank bracketing structure. We further define specific rules for pronouns. For example, the "ana" tag in the BulTreeBank [10] is very important for pronoun phrases of the following kind:

```
<Pron><w ana="P...">....
```

First, in case the fourth position is filled by a "t" and the tag starts with "Ps" as in "Ps*t*"[4], this is a *possessive form* and is part of the NP phrase in the transformation structure. For example:

хубавата ми кола

```
'beautiful-the my-clitic car'

my beautiful car
```

Or:

майка ми

```
'mother my-clitic'

my mother
```

Second, if the tag does not start with "Ps" then the pronoun is part of the verb phrase because it is a personal pronoun.

Finally, if there is no "t" on position four, but there is "l" or "-" instead, we annotate this as an NP (see the example above).

We further reduced the original BulTreeBank tagset [10] to a much smaller one with just 95 tags. In most cases, this meant losing some of the surface morphological forms. For example, in the example sentence above, the word "си" ("our own-clitic") was originally annotated with the tag "Psxt" (pronoun, possessive, reflexive, short form), which was transformed to "PP$" (pronoun, possessive). Similarly, the last word in the example sentence, "предизвикателства" (challenges), had the tag "Ncnpi" (noun, common, neuter, plural, indefinite), which was collapsed to "NNS" (noun, plural). In other cases, the tags were directly transformed, e.g., the word "от" (from) with tag "R" (preposition) was transformed to the tag "IN" (preposition or subordinating conjunction).

3 Experiments and evaluation

In our experiments, we used the training and the testing sections of the BulTreeBank [5, 9] without further modifications in order to retrain for Bulgarian the sentence splitter, the tokenizer, the POS tagger, the chunker (shallow parser), and the syntactic parser from the OpenNLP tools. The results are shown in Table 1. The sentence splitter achieved an F_1 score of 92.54%. The false positives constituted most of the errors and appeared in complicated sentences rather than at abbreviations of organization names as we expected. For example, the following chunk was recognized as a sentence:

Кой беше този човек?, би запитал той. Така че...

```
Who was that guy?, he would ask. So ...
```

However the actual sentence should have been:

Кой беше този човек?, би запитал той.

```
Who was that guy?, he would ask.
```

Some errors appeared in sentences annotated with direct speech, e.g., the sentence tagger annotated the following piece of text as a sentence:

Наведен напред, Той впери поглед в мрака
и рече: - Да бъде светлина.

```
Inclined forward, he took a look in the dark
 and said: - Let there be light.

Inclined forward, he stared at the gloom
 and said: - Let there be light.
```

However, the actual sentence in the BulTreeBank was the following one:

Наведен напред, Той впери поглед в мрака
и рече:

```
Inclined forward, he took a look in the dark
and said:

Inclined forward, he stared at the gloom
and said:
```

The tokenizer achieved an F_1 score of 98.49%. The majority of the errors appeared in sparse abbreviations. For example, the abbreviation for kilograms, "кг.", was frequently tokenized as "кг". The abbreviation for vehicle "horse power", "к.с.", was wrongly tokenized as "к.с". Some errors involved words that contained no space, e.g., "предсказание" (prediction), which were wrongly tokenized as "казание" (an old Bulgarian word that means "statement"). There were also some rare errors in names of people and locations, e.g., the name "Лазаръс" (Lazarus) was tokenized as the Bulgarian name "Лазар" (Lazar), and the city "Казанлък" (Kazanlak) was tokenized as "Казан" (Kazan).

The POS tagger achieved an F_1 score of 90.34% on the full morpho-syntactic tagset of the BulTreeBank. We made use of the tagger's ability to employ a tag dictionary that has been automatically generated from the training data and used internally to enumerate over a fixed set of possible tags for each word as opposed to allowing all possible tags; this severely limited

[4] Here * stands for a character that is not important for the discussion.

Sentence splitter	Tokenizer	POS Tagger	Chunker	Parser
92.54	98.49	94.43	84.60	77.56

Table 1: ***The F_1 scores (in %) of the OpenNLP components discussed in the study.***

the number of decision options available per word. The majority of the erroneous morpho-syntactic tags were not entirely wrong; rather, only some of their components were incorrect. For example, "чорбаджиите" was wrongly annotated as "noun, common, feminine, plural, definite" while the correct tag was "noun, common, masculine, plural, definite".

```
wrong annotation
‘‘чорбаджиите’’ (gaffer) with POS=Ncfpd

correct annotation
‘‘чорбаджиите’’ with POS=Ncmpd
```

Another common error was the wrong annotation of proper nouns as common nouns. Here is an example for the person name "Странджата" (*Strandzhata*, a literary character):

```
wrong annotation
‘‘Странджата’’ with POS=Ncfsd

correct annotation
‘‘Странджата’’ with POS=Npfsd
```

Another location example was the word "Балканът" (*Balkanut*, i.e., the Balkan mountain):

```
wrong annotation
‘‘Балканът’’ with POS=Ncmsd

correct annotation
‘‘Балканът’’ with POS=Npmsd
```

We further collapsed the original BulTreeBank tagset, which contained 680 morpho-syntactic tags, to a much smaller one, with 95 tags only; see Section 2 for details. The POS tagger with that reduced tagset achieved an F_1 score of 94.43% and was used in the experiments on chunking and syntactic parsing described below. For comparison, the best results for English on the Penn Treebank are 97.33% [8], but using a different learning algorithm: bidirectional perceptron.

The chunker achieved an F_1 score of 84.60%. This result is much lower than the best F_1 score for English reported at the CoNLL-2000 chunking competition: 94.13%. However, this comparison should be treated with caution since we did no special adaptation of the features to Bulgarian. We should also note that the ChunkLink[5] script, used at CoNLL-2000, was tailored to the Penn Treebank tagset, and was thus not very suitable to our collapsed BulTreeBank tagset.

The syntactic parser: Unfortunately, we were unable to evaluate our parser with the full morpho-syntactic tagset of the BulTreeBank; this would have required coding efforts for some parts of speech, e.g., nouns, that go beyond simple adaptation. On our collapsed tagset, we achieved an F_1 score of 77.56%. For comparison, the best parser for English that only uses Penn Treebank data achieves F_1=91.4% [3].

4 Conclusions and future work

We have described our efforts in adapting five basic NLP components from English to Bulgarian using manually annotated training data from the BulTreeBank. We have further presented the first systematic evaluation of NLP components for Bulgarian on the same dataset and within the same machine learning framework: maximum entropy. The evaluation results have shown an F_1 score of 92.54% for the sentence splitter, 98.49% for the tokenizer, 94.43% for the part-of-speech tagger, 84.60% for the chunker, and 77.56% for the syntactic parser, which should be interpreted as baseline for Bulgarian.

In future work, we will improve on the above baseline results, e.g., by adding language-specific features. We further plan to adapt two other maximum entropy based OpenNLP components to Bulgarian: for named entity recognition and for co-reference resolution.

Acknowledgments

The work reported in the paper is partially supported by the EU FP7-212578 project LTfLL – Language Technologies for Life Long Learning.

References

[1] A. Berger, S. D. Pietra, and V. D. Pietra. A maximum entropy approach to natural language processing. *Computational Linguistics*, 22:39–71, 1996.

[2] E. Buyko, J. Wermter, M. Poprat, and U. Hahn. Automatically adapting an NLP core engine to the biology domain. In *Joint BioLINK-Bio-Ontologies Meeting*, pages 65–68, 2006.

[3] E. Charniak and M. Johnson. Coarse-to-fine n-best parsing and maxent discriminative reranking. In *ACL'05: Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*, pages 173–180, Ann Arbor, Michigan, June 2005.

[4] M. P. Marcus, B. Santorini, and M. A. Marcinkiewicz. Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics*, 19, 1993.

[5] P. Osenova and K. Simov. BTB-TR05: BulTreeBank Stylebook. Technical report, BulTreeBank Project Technical Report, 2004.

[6] A. Ratnaparkhi. *Maximum Entropy Models for Natural Language Ambiguity Resolution.* PhD thesis, University of Pennsylvania, 1998.

[7] F. Sha and F. Pereira. Shallow parsing with conditional random fields. In *NAACL'03: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pages 134–141, 2003.

[8] L. Shen, G. Satta, and A. Joshi. Guided learning for bidirectional sequence classification. In *ACL'07: Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 760–767, Prague, Czech Republic, June 2007.

[9] K. Simov. BTB-TR01: BulTreeBank Project Overview. Technical report, BulTreeBank Project Technical Report, 2004.

[10] K. Simov, P. Osenova, and M. Slavcheva. BTB-TR03: BulTreeBank Morphosyntactic Tagset. Technical report, BulTreeBank Project Technical Report, 2004.

[5] `http://ilk.kub.nl/~sabine/chunklink`