### Feature-Rich Part-Of-Speech Tagging Using Deep Syntactic and Semantic Analysis

Luchezar Jackov Institute for Bulgarian Language Bulgarian Academy of Sciences lucho@skycode.com

#### Abstract

This paper describes the implementation, improvement and evaluation of the machine translation (MT) system proposed by Jackov (2014) when used as a feature-rich part-ofspeech (POS) tagger for Bulgarian. The system does not rely on POS tagging for morphological disambiguation. Instead, all ambiguities are considered in parsing hypotheses that are scored and the best one is used for tagging. The system does not use automatic training on annotated corpora. Manually and automatically compiled linguistic resources are used for hypothesis derivation and scoring. BulTreeBank manually annotated corpus (Simov and Osenova, 2004) was used for evaluation, error detection and improvement.

#### 1 Introduction

Part-of-speech (POS) tagging is the activity of labeling the words of a text with contextual tags describing the various grammatical features of the specific word usage. This is not trivial since many word forms are homonymous to other word forms. For instance, "*water*" is a noun in "*I drink water*" and a verb in "*They water the gar-den*". Linguists normally classify the words into at least eight basic POS classes: noun, pronoun, adjective, verb, adverb, preposition, conjunction, and interjection. Sometimes the list is extended with numerals, determiners, particles, etc. but the number of classes rarely exceeds 15.

Computational linguistics works with a larger inventory of POS tags, e.g., the Penn Treebank (Marcus et al., 1993) uses 48 tags: 36 for part-ofspeech, and 12 for punctuation and currency symbols. The increase in the number of tags is partially due to finer granularity, e.g., there are special tags for determiners, particles, modal verbs, cardinal numerals, foreign words, existential *there*, etc., but also to the desire to encode morphological information as part of the tags. POS tagging poses major challenges for morphologically complex languages whose tagsets encode a lot of additional morpho-syntactic features (for most of the basic POS categories), e.g., gender, number, person, etc. For example, Bul-TreeBank (Simov and Osenova, 2004) for Bulgarian uses 680 tags, while the Prague Dependency Treebank (Hajič, 1998) for Czech has over 1,400 tags (Georgiev et al., 2012).

POS tagging is a form of disambiguation and in many cases a deep syntactic and semantic analysis is needed for correct tagging.

An interesting approach for deep syntactic and semantic disambiguation was presented by Jackov (2014). However, the paper indicated that no evaluation of the system has been made. The goal of this paper is to present an evaluation of this system by using it as a feature-rich morphological tagger for Bulgarian and comparing the system output to the BulTreeBank manually annotated corpus for Bulgarian (Simov and Osenova, 2004).

The proposed approach considers the input text as a sequence of tokens. Then for each token all possible lemmas are derived. Lemma sequences of 1 or more tokens are looked up by the concept binder module in a synset lexicalization table for WordNet (Fellbaum, 1998) synsets. Each successful look-up is an assumption for a concept and constitutes an initial parsing hypothesis. The hypotheses contain assumptions about the concepts lying behind the input tokens, their syntactic roles and their dependency relations. Adjacent hypotheses are combined into new hypotheses for larger spans of the input sequence by using manually written hypothesis derivation rules. Each rule identifies, inherits and extends the syntactic and semantic assumptions of the constituting hypotheses. The rules are applied using a modified version of the Cocke-Younger-Kasami (CYK) algorithm (Cocke et al., 1970; Younger, 1967; Kasami, 1965) until all spans of the input sequence are covered. To prevent hypothesis space explosion each hypothesis is scored against a knowledge database of dependency relations and only the n-best hypotheses are kept for each span of tokens.

Every hypothesis identifies one lemma per token and the best hypothesis is used for the tagging task. The data for the lemma consists of a set of values of morphological categories such as part of speech, gender, number, article, case, etc. These attribute values are used to compile the morphological tag assigned to each token.

The system was improved by correcting the handling of family names, by adding a category for explicit marking of verb transitiveness and importing verb transitiveness data from a dictionary, and by extending its lexical database using BulNet (Koeva, 2010).

The rest of the paper is organized as follows: Section 2 provides an overview of related work, Section 3 describes Bulgarian morphology in brief, Section 4 provides detailed description of the system, Section 5 describes modification of the system for the POS tagging task, Section 6 presents the work on the evaluation of the system and its improvement by using additional resources, Section 7 discusses in detail the process of error analysis and the resulting improvement, and Section 8 concludes and describes some promising directions for future work.

#### 2 Related Work

A comprehensive review of the recent research on POS tagging is given by Georgiev et al. (2012). The rest of the paragraph is provided from the above-mentioned paper for informative purposes. Most previous work on Bulgarian POS tagging has started with large tagsets, which were then reduced. For example, Dojchinova and Mihov (2004) mapped their initial tagset of 946 tags to just 40, which allowed them to achieve 95.5% accuracy using the transformation-based learning of Brill (1995), and 98.4% accuracy using manually crafted linguistic rules. Similarly, Georgiev et al. (2009) who used maximum entropy and the BulTreeBank (Simov and Osenova, 2004) grouped its 680 fine-grained POS tags into 95 coarse-grained ones, and thus improved their accuracy from 90.34% to 94.4%. Simov and Osenova (2001) used a recurrent neural network to predict (a) 160 morpho-syntactic tags (92.9% accuracy) and (b) 15 POS tags (95.2% accuracy). Some researchers did not reduce the tagset: Savkov et al. (2011) used 680 tags (94.7% accuracy), and Tanev and Mitkov (2002) used 303 tags and the BULMORPH morphological analyzer (Krushkov, 1997), achieving P=R=95%. (Georgiev et al., 2012)

Chanev and Krushkov (2006) have also done a preliminary research on using HMM for POS tagging for Bulgarian, achieving precision of 92.16%.

A combined method for POS tagging, dependency parsing and co-reference resolution for Bulgarian has been proposed in Zhikov et al. (2013). The approach of Jackov is similar to the above-mentioned method in obviating the POS tagging step and the simultaneous resolution of all the morphological ambiguities together with the syntactic and semantic ambiguities. However, all of the linguistic data it uses is defined explicitly and only the dependency relations knowledge database may be automatically populated, while most of the other approaches rely on machine learning taking arbitrary features from training datasets. The predefined linguistic data is used to generate and score the hypotheses for the input sequence, eventually using the best hypothesis for output.

### **3** Bulgarian Morphology

Bulgarian language is highly inflective and with very rich morphology. Some of the pronouns have more than ten grammatical features, including case, gender, person, number, definiteness, etc.

There is a number of lexical and grammatical ambiguities in Bulgarian. For instance, many Bulgarian verbs have the same form for 2-nd and 3-rd singular aorist or imperfect, e.g. *Ade(ue) πu?* (meaning 'Did you/he eat'). There are also cross-POS ambiguities such as cmasa, which means (a) 'joint' (a noun) or (b) 'become' (a verb, 3-rd person singular present). There is a systematic ambiguity between adverbs and neuter singular adjectives which all have the same surface form, e.g. бързо is an adverb in Той кара бързо (meaning 'He drives fast') and an adjective in бързо хранене (meaning 'fast food'). Note that the example given in English has the same ambiguity. There is another notable ambiguity for the possessive clitic pronouns and the dative clitic personal pronouns. The situation is even worse for the conjunction u (meaning 'and') and  $\dot{u}$ , which is the clitic form of the possessive pronoun (meaning 'her') and the dative clitic form of the personal pronoun ma (meaning 'she'). Note that in the real world  $\dot{u}$  is often written without

the stress mark, which makes it identical to the conjunction u.

An analysis of BulTreeBank shows that it consists of 59,924 different morphological entities (a word form and its morphological tag). 52,017 of them are unambiguous in terms of tagging, i.e. they are tagged the same within the corpus. However, the ambiguous word forms prevail in terms of usage statistics.

#### 4 Detailed Description of the System

### 4.1 Overview

The system has been implemented in C++ and has a very compact binary data representation, approx. 60MB for 7 languages and 42 language translation directions. It has been used in offline translation applications for mobile devices, outperforming Google Offline Translator in both quality and size (the latter needs about 1.05GB of data for the above-mentioned 7 languages). It has also participated successfully in the iTranslate4 project, and can be tested online at http://itranslate4.eu (the SkyCode vendor). The system consists of a lemmatizer, a concept binder, a hypothesis generator, a dependency relations scorer and a synthesis unit. (Jackov, 2014)

The system implements an extensive inventory of categories and category values. A special category, the hypothesis type identifier (HTI), serves as the set of non-terminal values for the parsing rules, which are extended context-free grammar (CFG) rules used for production of hypotheses.

An elaborate description with many more examples is given by Jackov (2014).

#### 4.2 Lemmatizer

The first step of the system operation is to apply the lemmatizer module on each input token, which produces a list of all lemmas for each token along with their category values. For instance, for the input token *mu*, the module will produce an entry for the dative clitic of the personal pronoun a3 (meaning 'I'), an entry for the possesive pronoun clitic and two more entries for the second and third person singular aorist forms of the verb *mus* (meaning 'to wash'). The lemma of each lemmatization is kept as a lemma identifier, which is used later in the concept binder. The lemmatizer is built as a simple, yet very efficient stemmer allowing definition of arbitrary paradigms, one per HTI. The original system has 102,393 lemmas for Bulgarian.

#### 4.3 Hypothesis Generator

The second step is to apply the hypothesis generator for every span of the input sequence of tokens. The module first runs the concept binder for spans of length less than 7 tokens, and then applies parsing rules over the adjacent sub-spans of each span.

#### 4.4 Concept Binder

The concept binder finds the concepts (WordNet synset identifiers) matching a span of input to-kens.

It uses a database of the possible lexicalizations for each WordNet synset. Each lexicalization entry in the database consists of a list of lemma identifiers, WordNet synset identifier, attribute restriction rules, attribute unification rules, and a list of additional attribute values. The list of additional values is used to define lexicalization level features such as sub-categorization frames, transitiveness and aspect for verbs, etc. The original system has 166,948 synset lexicalizations for Bulgarian.

### 4.5 Parsing Rules and Hypothesis Generation

The core of each parsing rule is an extended CFG rule defined for the HTI feature values of the constituting hypotheses. The parsing rule extends the CFG by defining additional attribute value restrictions, agreement restrictions, attribute unification rules and parsing rule score. It also defines syntactic and semantic roles, dependency relations and propagation rules so that the higher level hypothesis resulting from the rule application unifies those of the constituting hypotheses.

## 4.6 Dependency Relations Knowledge Database

The database contains entries that consist of a relation identifier, two WordNet synset identifiers and a weight value, which is normally 1 or -1.

The database is manually populated and currently has 1,803,446 entries.

Here are sample entries with words instead of WordNet synset identifiers for clarity:

(poss, study, woman, 1)

(nsubj, mushroom, study, 1)

The above entries are enough for disambiguating the sentence *Women's studies mushroom*.

### 4.7 Hypothesis Scoring

As a result each hypothesis contains a number of assumed concepts and their dependency relations

and each concept is identified by its WordNet synset identifier. The set of the relations between the concepts is scored by looking up the dependency relations knowledge base. If the look-up is successful the dependency relation score is the weight of the matching entry, otherwise the score is zero. The hypothesis score is calculated by summing the dependency relation scores and the parsing rule score.

### 5 **POS Tagging by the System**

#### 5.1 Overview

When the hypothesis generator finishes its work it yields a parsing hypothesis for the input sequence of tokens having the best score. While the lemmatizer assigns all possible lemmatizations for each token, each hypothesis contains exactly one lemmatization per token. The lemmatization data kept by the system contains the feature values associated with the input token, which in turn are used to compile the POS tag that is ultimately assigned to the token.

#### 5.2 Translating Feature Values to Tags

The main issue when translating the feature values used within the system into the BulTreeBank tag set was the mapping of the large inventory of feature values (more than 1,000) into the large inventory of BulTreeBank tags (680).

Some of the work was easy due to the fact that the most common features and their values such as person, gender, number, etc. correspond in BulTreeBank and within the system of Jackov. For these features only a simple mapping of the feature values into the respective BulTreeBank mnemonic encodings and concatenating the resulting symbols was needed to correctly produce the BulTreeBank tags.

However, some of the word paradigms posed a problem. There are word forms in the lemmatizer that are handled by using derivation. The most notable examples are the verbal nouns, which are correctly annotated as nouns in Bul-TreeBank while being handled as derivational verb forms in the system. There are also others, e.g. various adjectives that are systematically derived from nouns and are handled as derivational noun forms within the system. Jackov motivated these deviations from the accepted linguistic models with much easier handling of such words within the system, their analysis, and translation. For instance, some of the derivational forms do not have WordNet synsets (at least in PWN3.0) as the verbal nouns have the verb semantics and the above-mentioned derivational adjectives in Bulgarian are semantically equivalent to English nouns used attributively.

#### 6 Evaluation and Improvement by Using Additional Resources

#### 6.1 Overview

A preliminary run of the system as a POS tagger for the BulTreeBank tagset produced result with accuracy of 88%. The error analysis showed the following deficiencies: (a) incomplete correspondence of category values to tags; (b) improper handling of family names; (c) lack of lemmas for some words; (d) lack of explicit transitiveness data in the lexicalization database; (e) lack of explicit adverb type description; (f) errors in the lemmatization data, the rules and the lexicalization data in the system. Handling these deficiencies is described in the below sub-sections.

#### 6.2 Improving the Handling of Family Names

The BulTreeBank tagset annotates family names using a special hybrid tag because family names in Bulgarian are inflected by gender and number. In the system family names are entered as proper nouns. This has been improved by defining a new HTI feature value and a respective paradigm for the word forms. A simple algorithm based on the word endings was applied to derive the paradigms of the family names that had been defined as proper nouns. It was based on the heuristic that most Bulgarian family names have unchanging suffixes from which the lemma (the singular masculine form) can be derived and the inflection group identifier can be assigned, after which the transformation is complete.

#### 6.3 Using BulNet

BulNet (Koeva, 2010) is the Bulgarian equivalent of Princeton WordNet (PWN) (Fellbaum, 1998). It is being developed by the Institute for Bulgarian Language (IBL) at the Bulgarian Academy of Sciences. The dataset was kindly provided by prof. Svetla Koeva from IBL.

A comparison between the dataset and the system lexicalization data showed that BulNet contained many lexicalizations that were not in the system and using BulNet will mitigate the deficiency of lacking some lexicalizations.

The use of the BulNet dataset was significantly eased by the fact that it uses the PWN 3.0 synset identifiers which are also used by the system.

## 6.4 Adding Explicit Transitiveness Feature Values

In the initial experiments the verb transitiveness was derived from the sub-categorization values that the system already had. However, this proved inconsistent with BulTreeBank. Apparently, the dictionary data for the transitiveness was used by the corpus annotators. To overcome this, an explicit transitiveness category has been added and the database has been populated with values by consulting the multi-volume Dictionary of Bulgarian Language by IBL.

## 6.5 Adding Explicit Adverb Type Feature Values

There is no adverb type categorization in the system, while most of the adverbs in BulTree-Bank are tagged along with a type value. Since there was no other source for deriving this information, the most commonly used adverb tags have been used to populate the system database with explicit adverb type category values.

# 6.6 Using Unambiguous Word Forms as a Constraint

Additional improvement was achieved by analyzing the BulTreeBank corpus and extracting the unambiguous word forms (word forms that have unambiguous annotation), and using them as a constraint. For instance, this obviated the need of translating the category values for many pronouns which are elaborately annotated within BulTreeBank. However, using this technique also hides some of the corpus errors that become evident when comparing the POS tagging output of the system to the corpus.

#### 6.7 Manual Improvement

After the above-mentioned improvements the precision of the POS tagging by the system reached 93%. The error analysis showed the following causes for errors: (a) improper correspondence of category values to tags; (b) improper rule application due to improperly defined constraints; (c) missing rules for certain linguistic phenomena; (d) improper or missing lexicalizations; (e) improper verb transitiveness and aspect data; (f) improper paradigm definitions; (g) tagging errors in the corpus.

Trying to address (a), (b), (c), (d), (e), and (f) for just one of the corpus files improved the overall precision to 95%, and the precision of the POS tagging for that file reached 96.54%.

Further analysis of the errors showed that some of them were indeed annotation errors in the corpus, while others come from different strategies for handling specific language phenomena. instance. много For (meaning 'many/much/very') is always annotated as adverbial numeral in BulTreeBank which does not reflect the ambiguity of the word – it is a numeral when meaning *many*, a quantifier adjective for one of the meanings of *much*, and an adverb for another of the meanings of much and also an intensifier adverb meaning very. After contacting Kiril Simov and Petya Osenova, it became clear that this type of annotation is correct in terms of the annotation model they had accepted.

The system makes the above distinctions which results in POS tagging differences which however are not errors. After manually correcting the discrepancies in the corpus file and correcting other annotation errors, the tagging precision for that file reached 97.998%. The percentage of errors and discrepancies for the corrected version of the file when compared to the original corpus file was 1.722%.

### 7 Error Analysis and Improvement

The careful error analysis has lead to:

- improving the system where the cause was incorrect description of the linguistic phenomena;
- improving the corpus by correcting incorrect annotations where the cause was an annotation errors.

It is worth mentioning that the error analysis lead to discovering errors in all resources used by the system. However, the goal of this paper is to evaluate the system using BulTreeBank, that is why the errors found in other resources are not discussed.

## 7.1. Error Analysis Leading to Improvement of the System

Some of the most useful cases of error detection and correction that lead to the most significant improvements of the system were those of a missing rule or improper constraint definitions for a certain rule. The lack of constraints in the rule definitions results in generation of parsing hypotheses that are not grammatical, which in turn leads to incorrect tagging.

Examples of linguistic phenomena that were not handled and were addressed by adding rules:

- repetitive coordinating conjunctions (e.g. κακmo ..., maκa u ..., meaning '... as well as ...');
- handling personal pronoun dative clitics in front of a passive construction, i.e. *Писмата му бяха дадени* (meaning 'the letters were given to him').

Examples of linguistic phenomena whose handling was corrected by refining the rule constraints are:

- the personal and possessive pronoun clitics may appear before the verb. For instance, mu My κακυ moba, meaning 'you tell this to him'. However, it is unacceptable to start a sentence in Bulgarian with such pronoun. That constraint was added to the respective rules after inspecting tagging errors of u that should have been tagged as a conjunction (meaning 'and') but was erroneously tagged either as a dative clitic or a possessive clitic (meaning 'her');
- An adverbial phrase can appear between the verb and the direct object. For instance, *moŭ npasu uecmo mosa*, meaning 'he often does this'. However, this is unacceptable when the direct object is a personal pronoun clitic.
- Possessive pronoun clitics may appear outside the noun phrase before the verb.
  For instance, не ми забравяй рождения ден, meaning 'Do not forget my birthday'. However, no other word (such as an adverb or adverbial phrase) is allowed between them.

The error analysis has also lead to a number of lexicalizations such as *играя театър* (meaning 'to pretend') being added to the concept database.

# 7.2 Error Analysis Leading to Improvement of the Corpus

Some of the differences between the corpus files and the POS tagging result of the system turned out to be annotation errors. Some of these errors were sporadic, while others appeared to be systematic. Below is a list of the most frequent systematic errors that were discovered:

• Errors in transitivity annotation. The transitivity of many Bulgarian verbs varies depending on the specific usage and often changes when the verb is used reflexively. 88 out of 148 tagging differences between the original and the edited corpus file are transitivity annotation er-

rors. The above statistics are only for the 8,590-token file that was exhaustively inspected.

- Inconsistent annotation of the tokens 2/dee/dea (meaning 'two'). These forms were annotated in a number of different ways. In 265 cases throughout the corpus 2 was annotated just as a numeral (M) without any other feature values. In 61 cases it was annotated as Mc-pi (plural cardinal numeral, no gender, indefinite article). In 90 cases it was annotated as Mcxpi (x stands for the various gender values and there were annotations for masculine, feminine and neuter). In all 291 cases dee (meaning 'two' for feminine and neuter gender) was annotated as Mc-pi. In all 202 cases *dea* (meaning 'two' for masculine gender) was annotated as Mcmpi. The same goes with other numerals ending in '2' where the linguistic expansion of the numeral would require the gender feature value.
- Inconsistent annotation of numerals representing years. Numerals like *19xx* most probably represent years. Most such numerals are either correctly annotated as Mofsi (ordinal numeral, feminine, singular, indefinite article) or incorrectly annotated just as a numeral (M) without any feature values. The distribution varies for the different numerals from 66% to 33% to more than 50% for the improper annotation (just M) for some of the year values (e.g. 1996).
- Inconsistent annotation of numerals representing days of the month. Days of month are normally annotated as ordinal numerals and a singular noun for the month. However, many numerals representing a day of month were annotated just as M.
- Inconsistent annotation of *vaca* (meaning 'o'clock'). The correct annotation for this word form should be Ncmsh (single masculine noun with hybrid article). In many cases it was incorrectly annotated as Ncmpt (plural masculine noun count form). This may be caused by the constraint mentioned by Georgiev et al. (2012) in section 4. Apparently this rule is not appropriate for the above linguistic phenomenon.
- Inconsistent annotation of the article when annotating abbreviations. For in-

stance,  $C \square C$  (meaning 'Union of democratic forces') in 248 cases is annotated as Npmsi (proper noun with indefinite article) and in 79 cases as a proper noun with definite article. The same goes for other abbreviations such as  $\mathcal{B}C\Pi$  (meaning 'Bulgarian socialist party'), and MBP(meaning 'Interior ministry'). While for some of the abbreviations it may be arguable whether to use definite article or not, in the case of  $C \square C$  it is more often than not, in contrast with the above numbers.

• Inconsistent tagging of the auxiliary particle  $\partial a$  as an affirmative particle in one of the corpus files. This error alone accounts for 2% error rate in the annotation of that file.

#### 8 Conclusion and Future Work

#### 8.1 Conclusion

The experiments of evaluating the system of Jackov as a feature-rich POS tagger for Bulgarian proved to be useful in several ways. After inexhaustive manual improvement the precision for one training file reached the state-of-the-art value of 97.98% for the full BulTreeBank tagset (Georgiev et al., 2012) and exceeded the value of 97.13% for the partially similar approach of Zhikov et al. (2013). The precision for a reduced set of 13 tags reached 99.94%. The overall precision of over 95% (98.43% for 13-strong tagset) can also be considered very good, having in mind the high rate of over 1.7% of annotation disagreements and errors.

It is worth noting that the above precision rate is measured for the corrected version of the corpus, which makes the result not directly comparable to other results. However, the corrections made to the corpus are linguistically motivated and linguistically motivated corrections are needed for further progress (Manning, 2011).

The rule-based nature of the system makes it a valuable tool for discovery of annotation errors – nearly a third of the differences between the output of the system and the corpus turned out to be annotation errors.

#### 8.2 Future Work

The improvements made to the system in the process of using and refining it as a feature-rich POS tagger proved valuable as they improve the parsing accuracy and in turn the translation accuracy. A thorough review and corrective actions for POS tagging differences for all corpus files would (a) improve its parsing precision and translation quality and (b) improve the annotation precision of the corpus. It is also a good idea to evaluate the system using another POS-annotated corpus, e.g. BulPosCor<sup>1</sup> that was used by Dojchinova and Mihov (2004).

The good results and the improvement of the system in the process of evaluating it as a POS tagger imply that it is quite probable to achieve even better improvement and good results when evaluating it as a dependency parser by comparing its output to dependency-annotated corpora.

Another good direction of work is the use and evaluation of the system for semantic disambiguation, for instance using BulSemCor<sup>2</sup>;

Some of the tagging errors imply that improving the co-referential resolution of the system may yield even better results when used as a POS tagger.

#### 9 Acknowledgments

I wish to express my gratitude to my scientific advisor prof. Svetla Koeva for her help and encouragement to do the research described in the paper. I wish to thank assoc. prof. Petya Osenova and assoc. prof. Kiril Simov for the readily provided corpus data and their valuable comments on the research and when writing the paper. I also wish to thank the reviewers and my colleagues who have read the various versions of the paper and provided me with valuable comments.

#### References

- Eric Brill. 1995. Transformation-based error-driven learning and natural language processing: a case study in part-of-speech tagging. *Comput. Linguist.*, 21:543-565.
- - TAGGER\_FOR\_BULGARIAN
- J. Cocke, and J. T. Schwartz. 1970. Programming Languages and Their Compilers: Preliminary Notes. Technical report. Courant Institute of Mathematical Sciences, New York University.

<sup>&</sup>lt;sup>1</sup> http://bg.wikipedia.org/wiki/БулПосКор

<sup>&</sup>lt;sup>2</sup> http://bg.wikipedia.org/wiki/БулСемКор

- Veselka Dojchinova and Stoyan Mihov. 2004. High performance part-of-speech tagging of Bulgarian. In Christoph Bussler and Dieter Fensel, editors, AIMSA, volume 3192 of *Lecture Notes in Computer Science*, pages 246-255. Springer.
- C. Fellbaum. 1998. WordNet: An Electronic Lexical Database. MIT Press.
- Georgi Georgiev, Preslav Nakov, Petya Osenova, and Kiril Simov. 2009. Cross-lingual adaptation as a baseline: adapting maximum entropy models to Bulgarian. In *Proceedings of the RANLP'09 Workshop on Adaptation of Language Resources and Technology to New Domains*, AdaptLRTtoND '09, pages 35-38, Borovets, Bulgaria.
- Georgi Georgiev, Valentin Zhikov, Kiril Ivanov Simov, Petya Osenova, and Preslav Nakov. 2012. Feature rich part-of-speech tagging for morphologically complex languages: Application to Bulgarian. In *EACL'12*, pages 492-502.
- Jan Hajič. 1998. Building a Syntactically Annotated Corpus: The Prague Dependency Treebank. In Eva Hajičova, editor, *Issues of Valency and Meaning*. *Studies in Honor of Jarmila Panevova*, pages 12-19. Prague Karolinum, Charles University Press.
- Luchezar Jackov. 2014. Machine translation based on WordNet and dependency relations, In *Proceedings of Computational Linguistics in Bulgaria* 2014, pages 64-72.
- T. Kasami. 1965. An Efficient Recognition and Syntax-analysis Algorithm for Context-free Languages. *Scientific report AFCRL-65-758*. Bedford, MA: Air Force Cambridge Research Lab.
- Svetla Koeva. 2010. Bulgarian Wordnet current state, applications and prospects. In *Bulgarian-American Dialogues*, pages 120-132, Sofia: Prof. M. Drinov Academic Publishing House.
- Hristo Krushkov. 1997. *Modelling and building machine dictionaries and morphological processors (in Bulgarian)*. Ph.D. thesis, University of Plovdiv, Faculty of Mathematics and Informatics, Plovdiv, Bulgaria.
- Christopher D. Manning. 2011. Part-of-Speech Tagging from 97% to 100%: Is It Time for Some Linguistics? In Alexander Gelbukh (ed.), Computational Linguistics and Intelligent Text Processing, 12th International Conference, CICLing 2011, Proceedings, Part I. Lecture Notes in Computer Science 6608, pp. 171-189. Springer.
- Mitchell P. Marcus, Mary Ann Marcinkiewicz, and Beatrice Santorini. 1993. Building a large annotated corpus of English: the Penn Treebank. *Comput. Linguist.*, 19:313-330.
- Aleksandar Savkov, Laska Laskova, Petya Osenova, Kiril Simov, and Stanislava Kancheva. 2011. A

web-based morphological tagger for Bulgarian. In Daniela Majchrakova and Radovan Garabik, editors, *Slovko 2011. Sixth International Conference. Natural Language Processing, Multilinguality*, pages 126-137, Modra/Bratislava, Slovakia. Helmut Schmid. 1994. Probabilistic part-of-speech

- Kiril Simov and Petya Osenova. 2004. BTB-TR04: BulTreeBank morphosyntactic annotation of Bulgarian texts. Technical Report BTB-TR04, Bulgarian Academy of Sciences.
- Hristo Tanev and Ruslan Mitkov. 2002. Shallow language processing architecture for Bulgarian. In *Proceedings of the 19th International Conference on Computational Linguistics,* COLING '02, pages 1-7, Taipei, Taiwan.
- D. H. Younger. 1967. Recognition and parsing of context-free languages in time n3. *Information and Control 10 (2)*: 189-208.
- Valentin Zhikov, Georgi Georgiev, Kiril Simov, Petya Osenova, 2013. Combining POS Tagging, Dependency Parsing and Coreferential Resolution for Bulgarian. *RANLP 2013*: 755-762.