Computer Science and Information Systems 15(3):799–820 https://doi.org/10.2298/CSIS180310027B

Learning Syntactic Tagging of Macedonian Language*[1]

Martin Bonchanoski[1], and Katerina Zdravkova[2]

[1] University Ss Cyril and Methodius, Faculty of Computer Science and Engineering
1000 Skopje, Macedonia
Dublin, Ireland
martinboncanoski@gmail.com
[2] University Ss Cyril and Methodius, Faculty of Computer Science and Engineering
1000 Skopje, Macedonia
katerina.zdravkova@finki.ukim.mk

Abstract. This paper presents the creation of machine learning based systems for Part-of-speech tagging of Macedonian language. Four well-known PoS tagger systems implemented for English and Slavic languages: TnT, cyclic dependency network, guided learning framework for bidirectional sequence classification, and dynamic features induction were trained. Orwell's novel "1984" was manually tagged from the authors and it was used split into training and test set. After the training of the models, a comparison between the models was made. At the end, a POS tagger with an accuracy that reaches 97.5% was achieved, making it very appropriate for the future grammatical tagging of the National corpus of Macedonian language, which is currently in its initial stage. The Part-of-speech tagger that was create is published online and free to use.

Keywords: Part-of-speech tagging, TnT tagger, Cyclic dependency network, Guided learning for bidirectional sequence classification, Dynamic features induction

1. Introduction

Grammatical tagging, or morphosyntactic annotation of text corpora is the process of associating labels or tags to each word token in the text, in order to indicate its grammatical classification [1]. It includes interpretative information to text corpora, providing information about the part-of-speech (POS), morphological and grammatical features of the words [2]. The morphosyntactic annotations attached to a segment do not refer to other segments or annotations, however the choice of an annotation may significantly depend on the surrounding context [3].

The importance of standardized language management, initiated to enable the creation and application of uniform treatment in the multilingual information society

* This is an extended version of the conference paper "Machine learning-based approach to automatic POS tagging of Macedonian language", published in the Proceedings of the 8th Balkan Conference in Informatics in September 2017.

[1] Macedonian language is included in the UN standardization of geographical names since 1977 (https://digitallibrary.un.org/record/29404/files/E_CONF.69_4%5BV.II%5D-EN.pdf)

was recognized by ISO/TC 37/SC series of standards, intended to define: principles and methods; terminology workflow and language coding; management of terminology resources; language resource management; as well as translation, interpreting and related technology [4]. With this long-lasting, stable and unified approach, current information management will definitely enhance the industrial, technical, and scientific exchange and implementation of an efficient language communication.

In spite of the efforts to standardize language management, there is still not a standard annotation schema. In the early 1990s, the large Brown corpus consisting of almost 5 million words of American English was created [5]. It was initially annotated with the POS Brown tagset, which recognized 87 simple tags [6]. The great advantage of this tagset was the possibility to tag compound words, extending the tagset with additional 100 compound tags [5]. Using the same corpus, the well-known Penn Treebank was created. It has a very simplified annotation schema consisting of 36 POS tags and 12 other tags [7]. Major advantages to Brown annotation are in its stochastic orientation, syntactic bracketing established by using predicate-argument structure, and in the disfluency annotation, which distinguishes complete utterances from incomplete ones, and labels a range of non-sentence elements [7]. Although intended for annotation of English language, it was widely used for other languages, including Chinese [8] and Arabic [9].

Very popular among NLP researchers is also the Lancaster-Oslo/Bergen (LOB) Corpus. Published in 1961, it consisted of approximately 1 million words of British English, representing only 40000 lemmas tagged with 97 syntactic tags [10, 11]. Created as an extension of Intex, Silberztein created the linguistic annotation system NooJ, which is very popular because of the plethora of various tools, and the simplicity of the tagsets [12].

Slavic languages have more complex syntax than English, so most of them employ their own, language specific annotation schemas. National Corpus of the Russian Language consists of several smaller corpora, including the syntactic corpus SynTagRus, with about 860000 words [13]. It is syntactically tagged with various annotation systems, including Mystem, and the language-independent system MaltParser [14].

Probably the most explored Slavic system is the Prague Dependency Treebank, which comprises three layers of annotation: morphological POS tagging, analytic syntactic tagging, and tectogrammatical tagging [15]. Its exhaustiveness was adopted for other languages, such as the Arabic [16] and Basque [17].

National Corpus of Polish language has a one-million word subcorpus that was manually morphologically annotated with Morfeuzs [18, 19].

The initiative to produce harmonised language resources for several East-European languages ended up with the Multext-East corpus of sentence aligned translations of the Orwell's "1984" [20]. The carefully established morphosyntactic descriptions have evolved significantly, reaching its fourth version, which encompasses all the Slavic languages [21]. They are used for world-level tagging of many South Slavic languages, including Slovene [22], Croatian [23], and Serbian [24].

Most of the current methods in human language technology research heavily rely on the use of linguistically annotated corpora. Text corpus annotation is costly and time-consuming process, therefore annotated text data are unavailable or annotation hasn't been performed for a number of languages. Macedonian is still one of them, although the effort to create an annotated corpus started more than a decade ago [25].

Unfortunately, the initial enthusiasm of several computer researchers had never got an appropriate support by the linguists, making the whole process slightly unreliable. Meanwhile, new interpretative dictionary was created [26]. First version of the electronic lexicon of Macedonian language was generated by Aleksandar Petrovski, and it became freely available for research purposes [27]. Produced using Nooj, its morphosyntactic descriptions (MSDs) were automatically mapped to Multext-East MSDs, resulting in sometimes incomplete annotation. Digital dictionary of Macedonian language [28] is yet another very valuable resource, particularly because it contains many examples, which facilitate manual POS tagging and annotation. With the new background, the second attempt to syntactically annotate the same corpus, which is presented in this paper is more mature and fruitful.

This paper presents the creation of a fully automated POS tagging of Macedonian language. It is an extension of the previous research done by the same authors, which resulted in a POS tagger with an accuracy of 96.37% [29]. New contribution encompasses both, the theoretical and the practical aspects. The previous error analysis suggested that broadening of the purely syntactic approach with some morphological information might further improve the result. Therefore, a study of the most popular systems for grammatical tagging, and the corresponding corpora they are related to, was made. Since Macedonian language belongs to Slavic languages, they were examined thoroughly as well. This study was introduced in the previous paragraphs.

The main sources of morphological ambiguity, divided into three clusters: adjectives vs. adverbs; verbs vs. nouns; and words with more than two tags were also analysed in more details. They are systematically illustrated with the examples from the Orwell's "1984" parallel Macedonian-English corpus. Their significant influence was confirmed by the error analysis. Experimental part was enriched with a new POS tagging technology, the dynamic feature induction which is the most accurate state of the art POS tagger of English language [30]. The presentation of the results is enlarged with the comparative analysis of the tools used for the POS tagging over the same corpus.

The rest of the paper is structured as follows: In Section 2 the process of manual syntactic annotation is presented. Additional attention is paid to POS ambiguities, their frequency and resolution. Section 3 describes the techniques that were used to build ML models. In the Section 4, results are presented, then a comparison to baseline tagger is made and an analysis of the tagging errors. The final section is a summarization of the research and announces future work and improvements that could be implemented.

2. Manual annotation of Macedonian corpus

2.1. Earlier annotation activities

Manual part-of-speech and grammatical annotation of Orwell's "1984" started in 2005, as part of the bilateral Macedonian-Slovene project "Gathering, Annotation and Analysis of Macedonian - Slovene Language Resources" financed by the Ministries of Science of both countries. In an absence of a digital version of the novel, it was OCR scanned using ABBYY FineReader [31] with Unicode/UTF-8 encoding. The scanned version included many spelling and encoding errors, which were manually corrected

during several annotation attempts. All the Macedonian morphosyntactic descriptors (MSDs) were defined and included in Multext-East version 3 [32]. A small spreadsheet-like tool was created to enable manual annotation of the corpus (Fig. 1). Unfortunately, for the sake of a better observability of the lexicon, MSD assignment was made without the context, resulting in an incomplete and in many cases, an incorrect POS tagging.

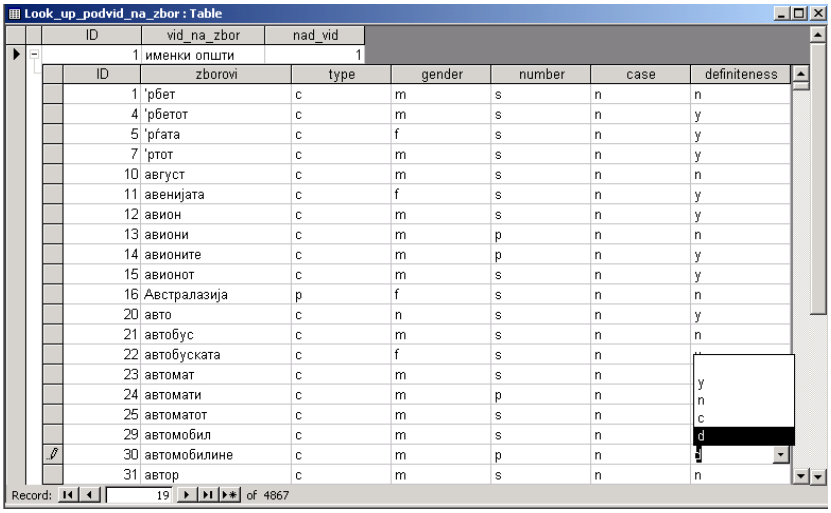

Fig. 1. The first tool for manual annotation of Orwell's corpus

The tool was very useful to extract and exhaustively annotate the nouns, verbs and adjectives from the novel, which were further used to estimate the capabilities of the system for learning first-order decision lists CLOG [25, 33]. The attempt of automatic POS tagging using the results of the manual annotation was very valuable, but again, rather incomplete [34]. Unfortunately, this tagger lacks manual disambiguation, instead the first available tag of a word was taken, which effectively removed all ambiguity. In the same period, Petrovski created the electronic lexicon of Macedonian language using NooJ morphosyntactic descriptions. NooJ MSDs were automatically mapped to more detailed Multext-East MSDs, which generated additional deficiencies in the annotated lexicon [35].

Recently, another independent attempt to automatically POS tag the same corpus was done [36]. Tagging was based on the POS tagged versions of Bulgarian, Czech, Slovene, English and Serbian language, which were pair-aligned to Macedonian translation. By using the cross-linguistic majority vote approach, they built a POS tagged version of the Macedonian corpus. Since Macedonian translation has a very consistent sentence and word alignment with the English original, the achieved accuracy of 88% is rather high. Unfortunately, the results of this very successful and valuable attempt for the automatic POS tagging of Macedonian language is not publicly available, thus it was not useful for the syntactic tagging presented in this paper.

It's worth to mention that Petrushev (2013) created the Python library nlmk (can be found at https://github.com/petrushev/nlmk) which is Natural Language processing library for Macedonian language. This library is based on definite set of rules that from our analysis is incomplete - after training the tagger in the output, not every word is

associated with a tag. Therefore, we cannot report the accuracy of this tagger nor the author has reported it.

2.2. The dataset

Orwell's "1984" corpus is relatively small if compared to the corpora that have been used for grammatical tagging of other languages. It contains 92327 words presented with 6667 sentences (Table 1). The corpus is encoded in XML format according to the rules from Text Encoding Initiative, TEI P4 [35].

Table 1. Properties of the dataset.

Type	Frequency
Sentences	6667
Tokens	108617
Words	92327
One POS class	58544
Multiple POS classes	33510
Unknown POS class	273
Punctuations	16290

Multext-East morphosyntactic specifications consist of 12 word classes. In the descriptions for the existing languages, several language specific features for Macedonian were introduced, such as the proximal and distal suffixed article of nouns and adjectives, as well as the oblique of some masculine nouns, which have a common dative and accusative form. Such property is very peculiar for Macedonian, since the language doesn't have cases. After establishing the morphosyntactic descriptions, automatic POS tagging was done by combining the results of all three existing sources: the annotated words [25], the lexicon [27], and the digital dictionary [28]. It was noticed than more than 36% of the words have more than one possible word class, a task that had to be resolved efficiently prior to any attempt to automatize the POS tagging.

For the purposes of automatic POS tagging, noun category was divided into two categories: common and proper nouns. This feature is the first information of nominal MSDs. Furthermore, modal verbs as a category were omitted for compatibility purposes with Multext-East project. One class for the abbreviations and one class for all other words (mainly foreign words written with the Latin script, or words from Orwell's Newspeak) have been introduced, so in total 14 classes were used including the punctuations. The distribution of the word classes in the dataset after the manual disambiguation is given in Table 2.

Table 2. Word class distribution in the dataset.

Word class	Frequency
Nouns	19655
Verbs	15365
Pronouns	12412
Prepositions	11973

Adjectives	10184
Conjunctions	9383
Adverbs	7574
Particles	3710
Numbers	1707
Proper nouns	381
Abbreviations	74
Interjections	35
Other words	30

2.3. Ambiguities in the corpus

As mentioned earlier in the paper, more than 36% of all the words in the corpus can belong to at least two different word classes. Undoubtedly, the most frequent ambiguity originates from the neuter adjectives, which have the identical form as the adverbs ending with the character "*о*". In the corpus, 2703 adverbs, and 1992 have this property. Most of them appear in the corpus with both forms. Additionally, the third person in singular of the Macedonian verbs in Present Tense ending in "*а*" can have the same form as some nouns in singular. Although the frequency of the verbs and nouns ending in "*а*" is considerable in the corpus very few examples prove this claim. Finally, there are several words with more than two potential POS tags. The following subsections present the most frequent adjective / adverb, verb / noun and multiple word class examples collocated in the target Macedonian and in the original English context.

Adjectives vs. adverbs. The word form *брзо* (Latin: *brzo*) appears only once as an adjective, with the meaning *rapid*, which is an adjective as well (*: едно потреперување, **брзо** како затворање бленда на фотоапарат ... / : it was only a twitch, a quiver, **rapid** as the clicking of a camera shutter ...*). On the other hand, the adverb *брзо* is more frequent with 45 appearances in the basic form, and additional 8 as the comparative of the same adverb: *побрзо* (*pobrzo*). The examples: "*Таа се фрли на креветот, и **брзо** ... го крена здолништето / She threw herself down on the bed, and **at once** ... pulled up her skirt*"; "*Назад во својот стан тој **брзо** помина покрај телекранот… / Back in the flat he stepped **quickly** past the telescreen …"; "На секој од нив **брзо** беше фрлен пропишаниот ручек ... / On to each was dumped **swiftly** the regulation lunch ...*" and "*Да претпоставиме дека ќе решиме **побрзо** да се истрошуваме. / Suppose that we choose to wear ourselves out **faster**.*" confirm the assumption by Aepli et al. [36] that the word classes usually match in translation. Additionally, in most of these cases, POS tag can be easily determined by the nearest previous or following word: *брзо потреперување*\N, *брзо помина*\V, *брзо беше*\V, *ќе решиме*\V *побрзо*. The only exception is the position of the corresponding verb *крена* in the sentence: "*... и **брзо** ... го крена здолништето*", since it is too distant from the adverb, making its disambiguation much harder.

The appearance of the word *грдо* (*grdo*) is predominantly adjectival: (*... неговото лице со груби црти , толку **грдо**, а сепак толку цивилизирано ... / ... his blunt-featured face, so **ugly** and yet so civilized …; Неговото **грдо** лице се доближи, ... / His large **ugly** face came nearer, ...*), and only once adverbial (*... со тежок црн мустак и со **грдо** убави црти ... / ... with a heavy black moustache and **ruggedly** handsome*

features). Although the word class of the source words is preserved, only the neighboring context of *грдо лице*\N is useful for automatic POS tagging.

Both, the word *јасно* (*jasno*) and its antonym *нејасно* (*nejasno*) originating from the same root have a comparative form *појасно* (*pojasno*) and *понејасно* (*ponejasno*). They appear it the corpus 16 times as adjectives *(... кога стана* ***јасно*** *дека Винстон ја бара собата ... / ... when it was made* ***clear*** *that Winston wanted the room ...*; *Ова, помисли тој со* ***нејасно*** *гадење ... / This, he thought with a sort of* ***vague*** *distaste ...*), and 20 times as adverbs: "*Гледа премногу* ***јасно*** *и зборува премногу отворено. / He sees too* ***clearly*** *and speaks too plainly.*"*;* "*Во секој момент од животот на кој можеше* ***јасно*** *да се сети / In any time that he could* ***accurately*** *remember ...*"*;* "*Многу* ***понејасно*** *помислуваше на Џулија. / More* ***dimly*** *he thought of Julia.*". POS tags of the source and target words are completely compatible, and the disambiguation by the context is rather straight forward, except in the *стана*\V *јасно*, where according to the collocation, the first association for *јасно* is an adverb, rather than the correct word class, which is an adjective.

Similar behavior and almost equal frequency have the adjectives: *силно* (*silno*) and *живо* (*zhivo*), which are consistently translated as adjectives: *vivid* and *strong*, or as adverbs: *luridly* and *tightly* for *силно*; and as adjectives: *alive* and *vivid*, or adverbs: *vividly* and *fast* for *живо*. However, the disambiguation due to the context is almost impossible for the word *силно*, which is distant from the word it is associated to.

Verbs vs. nouns. The typical Macedonian verbs that can be confused with the nouns are: *бара* (*bara, seek*\V vs. *ponds*\N), *изгледа* (*izgleda, look*\V, *appear*\V vs. *appearances*\N, *outlooks*\N), *лета* (*leta, fly*\V vs. *two, three, several summers*\N), *нема* (*nema, doesn't have*\V, *possess*\V, *exist*\V vs. *mute*\N), *скока* (*skoka, jump*\V vs. *jumps*\N) and *стана* (*stana, stand up*\V vs. *apartments*\N). They all exist in the corpus as verbs only, some of them frequently, such as *изгледа* (104 times) and *нема* (153 times). The two words belonging to both word classes are the words *игра* (*igra*) and *мора* (*mora*).

Игра was found five times as a noun, including: "*—сето тоа за нив беше еден вид блескава* ***игра****. / —it was all a sort of glorious* ***game*** *to them.", and "Беше тоа еден вид* ***игра****. / It was a kind of a* ***dance****.".* The same word was found four times as a verb, such as in the sentences: "*Белиот* ***игра*** *и матира во два потега. / White to* ***play*** *and mate in two moves.*" and "*… човек признава дека си* ***игра*** *со реалноста; / one admits that one is* ***tampering*** *with reality;*".

The verb *мора* is the modal verb *must*, thus it is very frequent in the corpus: "***Мора*** *да е некаде околу тој датум, ... / It* ***must*** *be round about that date, ..."; "И двајцата* ***мора*** *да биле проголтани ... / The two of them* ***must*** *evidently have been swallowed up ...". The noun is a part of the compound noun ноќна мора (nokjna mora / nightmare): "Ноќната* ***мора*** *почна со оној прв удар по лакотот. / With that first blow on the elbow the* ***nightmare*** *had started.".* The nouns are usually preceded by adjectives, while the modal verb *мора* is followed by the particle *да* (*da*), which significantly facilitates POS tagging. All these examples again confirm the hypothesis that source and target word classes are usually the same [36].

Words with more than two POS tags: Many Macedonian words and lemmas are homonymous, and they can belong to several word classes. The most frequent in Orwell's "1984" are the words: *добро* (*dobro*) and *само* (*samo*).

The word *добро* was found only once as a noun (Вие владеете со нас за наше **добро**. / You are ruling over us for our own **good**.); 28 times as an adjective in various

forms, for example: "„Беше тоа **добро** бесење", рече Сајм сеќавајќи се. / 'It was a **good** hanging,' said Syme reminiscently.", and "... прашање на **добро** воспитување на секој што знае за тоа ... / … it was only **common** courtesy in anyone else who knew of it …"; and finally, 53 times as an adverb: "Дури тогаш можеше за првпат да ја види **добро** жената. / For the first time he could see the woman **properly**.", and "Седејќи во длабнатината и седејќи **добро** навален назадечки **...** / By sitting in the alcove, and keeping **well** back …". Source and target words classes were always consistently preserved in the translated novel.

The word *само* was found 360 times in total. Its least repeated occurrence was as a conjunction (6 times): "*Помислете **само** што сè мораат тие да поднесат. / **Just** think what THEY have to put up with.*" and "*Минатото не **само** што се менуваше, ... / The past not **only** changed, ...".* More frequent were the adjectives, mainly determined with the definite article suffix "*то*" ("*to*"): "*Не **само** вредноста на искуството, туку и **самото** постоење на ... / Not **merely** the validity of experience, but the **very** existence of external reality*" and "*Никогаш не сум бил вистински вклучен во **самото** создавање на јазикот. / I have never had anything to do with the **actual** construction of the language.*". The particle само occurs 91 times: "*Важна беше **само** Полицијата на мислите. / **Only** the Thought Police mattered.*" and "*Но, таквите работи се дознаваа **само** преку нејасни гласини. / But one knew of such things **only** through vague rumours*". Expectedly, the most frequent role of the word was adverbial (262 times), such as in: "… ***само** делумно свесен за она што го прави . / **only** imperfectly aware of what he was setting down.", and "... пред **само** четири години. / It was **only** four years since ...*". The subtle role of the word само completely contradicts the hypothesis of similar source and target word classes, and additionally makes the POS tagging extremely complex.

Apart from these words, many other were also considered as extremely ambiguous: *еден / едно (eden / edno), колку (kolku), нешто (nesthto), освен (osven), and што (shto)*, so during semi-automatic POS tagging, they were checked more thoroughly before making the final decision.

2.4. Semiautomatic POS tagging

Two separately hosted versions of the GUI tool for semiautomatic POS tagging were created, each intended for one of the both human taggers (Fig. 2.). The tool was introduced in 2017 [37]. All the unique words from the corpus, which existed in the electronic lexicon [27] and in the digital dictionary [28] in only one class have been used to assign all the possible tags to the words in the corpus. Then, the assigned tags were cross-checked with the lists of manually extracted nouns, adjectives and verbs [25]. Parallel tagging of the ambiguous words was done by two independent persons (the authors of this paper).

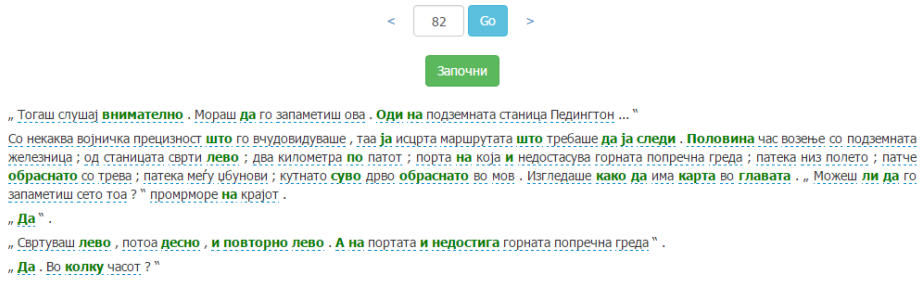

Fig. 2. Tool for manual annotation, resolved ambiguous word are colored in green

Both POS tagged texts were crossed-matched. The decision about those tags, which were not identical was again resolved using the second tool for mutual comparison of the preliminary results (Fig. 3.). As a result, an annotated corpora with all coarse POS tags for each word has been created. Even though the tagging wasn't done by linguists themselves, several distinguished linguists from the Institute of Macedonian Language "Krste Misirkov" have been consulted for the most common ambiguities.

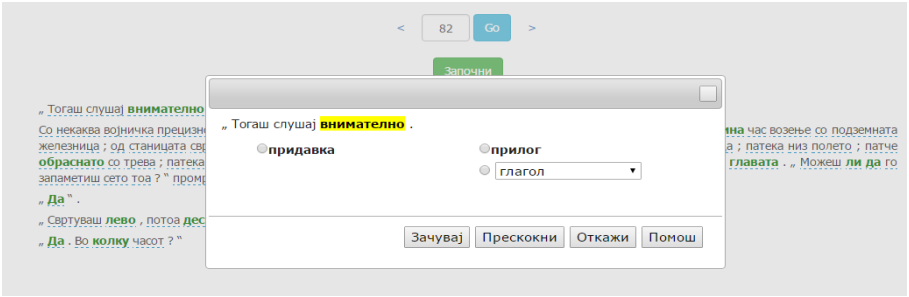

Fig. 3. The tool that compares POS tags done by two independent authors

3. Automatic tools for syntactic tagging

To enable the computer to automatically derive the POS category of the words in a given text, the language has to be modeled in some way. Features that capture the characteristics of the words and the context where they appear should be constructed and fed to a model that learns to predict POS tags for the new unknown words.

Many algorithms have been applied to this problem, including hand-written rules (rule-based tagging), probabilistic methods as well as other methods such as transformation-based tagging and memory-based tagging [38].

Since the rule-based tagging requires a lot of manual work and inclusion of language experts, this approach had to be discarded. Instead, the probabilistic methods and methods based on machine learning models were found more convenient for smaller corpora.

In total, four different methods were considered and tested. In contrast to the previous work, which included: TnT tagger [39], cyclic dependency network [40], and

guided learning for bidirectional sequence classification [41], an additional method was added. This new method is called NLP4J. It uses dynamic feature induction and it reports best results for English on WSJ corpus according to Association of Computational Linguistics (ACL) [42].

The methods were chosen by their popularity for English and Slavic languages, and by their availability to the authors of this paper. All methods that were chosen are robust and support tagging of unknown words.

In the following subsections, a brief overview of the technical specifications of each model is given. For deep-dive technical details of each of the models, the original papers are referenced.

3.1. TnT

TnT is the short name of Trigrams'n'Tags, which is an efficient statistical POS tagger [39]. The architecture is very flexible and it is applicable to a large variety of languages and almost any tagset. It was initially used by Brants, who proved that this tagger based on second order Markov model performs at least as well as the approach with Maximum Entropy framework.

This model is based on the second order Markov models. In its implementation, it contains the Viterbi algorithm [43]. The states of the model represent the POS tags, whereas the outputs represent the words. Transition probabilities depend on the states, which are a pair of tags. Output probabilities only depend on the underlying tag. They are calculated using the formula:

$$\underset{t1\ldots t_T}{argmax} \prod_{i=1}^{T} [P(t_i \vee t_{i-1}, t_{i-2}) P(w_i \vee t_i)]\, P(t_{T+1} \vee t_T). \quad (1)$$

for a given sequence of word $w_1 \ldots w_T$ of length T, where the tagset are denoted with t_1 to t_T.

Transition and output probabilities are estimated from a tagged corpus. TnT recursively uses unigrams, bigrams and trigrams to calculate the transition probabilities. They are presented with the formulas (2), (3), and (4) respectively:

$$\hat{P}(t_3) = \frac{f(t_3)}{N}. \quad (2)$$

$$\hat{P}(t_3|t_2) = \frac{f(t_2,t_3)}{f(t_2)}. \quad (3)$$

$$\hat{P}(t_3|t_1,t_2) = \frac{f(t_1,t_2,t_3)}{f(t_2,t_3)}. \quad (4)$$

Accordingly, the lexical probability is derived from the trigram probability as:

$$\hat{P}(w_3|t_3) = \frac{f(w_3,t_3)}{f(t_3)}. \quad (5)$$

Trigram probabilities generated from a corpus usually cannot directly be used because of the sparse data problem. Thus, smoothing paradigm is used. The smoothing

paradigm that delivers the best results in TnT is a linear interpolation of unigrams, bigrams, and trigrams:

$$P(t_3|t_1,t_2) = \lambda_1\hat{P}(t_3) + \lambda_2\hat{P}(t_3|t_2) + \lambda_3\hat{P}(t_3|t_1,t_2). \quad \textbf{(6)}$$

The sum of lambdas equals 1, thus *P* again represents a probability distribution. Their weights are calculated by deleted interpolation [38].

TnT handles unknown words by a suffix trie. Very often the word endings are a good indicator about the class of the word in many languages, which is also case in Macedonian. For example, suffixes: "*ски* (*ski*)", "*ска* (*ska*)", and "*ско* (*sko*)" indicate that the word form probably is an adjective, while: "*ние* (*nie*"), "*ање* (*anje*)", "*ење* (*enje*)", "*ство* (*stvo*)", "*ец* (*ec*)", "*ик* (*ik*)", "*тел* (*tel*)", and "*ина* (*ina*)" indicate that the word is a noun. With almost no exclusions, the suffix "*јќи* (*jkji*)" corresponds to adverbs, while "*ува* (*uva*)" is a typical sufix for verbs.

3.2. Maximum entropy cyclic dependency network

The previous model implements unidirectional approach to conditioning inference along the sequence, from left to right. But, the approach proposed by Toutanova implements explicit use of both preceding and following tag contexts via a dependency network representation with broad use of lexical features, including jointly conditioning on multiple consecutive words [40]. Furthermore, it makes effective use of priors in conditional loglinear models, and fine-grained modeling of unknown word features. It is not a standard Bayes network, because the graph has cycles. Rather, it is a more general dependency network. Other models that use unidirectional approach, use the left context of the target word explicitly. They also use the right context of the target word but implicitly. The right context is used when $P(t_{+1}|t_0,w_{+1})$ is calculated (where +1 denotes the word/tag in the next position).

However, it makes a lot of sense to use both contexts explicitly, as both of them can have the same impact on the decision what tag should be chosen. For example, in the sentence „*Цветовите се убави* (*Cvetovite are beautiful* / *Flowers are beautiful*)", the word „*се* (*se*)" is the third person plural of the auxiliary verb *е* (*to be*). However, the frequency of that word as a pronoun is a lot higher than as a verb in the corpus. Furthermore, the combination of a noun followed by a pronoun is very common because the reflexive pronoun „*се* (*se*)" very often is placed between a noun and a verb. But if the right context is used as well, then the combination verb – adjective is more common than pronoun – adjective, and it will result in a correct disambiguation.

In the cyclic dependency network that is used (Fig. 4.), each node represents a random variable along with a local conditional probability model of that variable, conditioned on the source variables of all incoming arcs. It looks like Bayes' net, but since there are cycles in the net, the chain rule that is used in Bayes' nets cannot be applied. Reconstructing the joint probabilities from these local conditional probabilities may be difficult, but estimating the local probabilities themselves is no harder than it is for acyclic models.

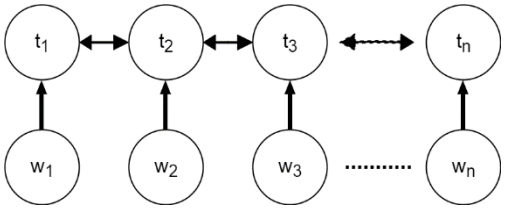

Fig. 4. Bidirectional dependency network, courtesy of Toutanova et al. [40]

The product of local probabilities from a dependency network can be seen as a product i.e. as a score:

$$score(x) = \prod_i P\left(x_i | Pa(x_i)\right). \quad (7)$$

where $P_a(x_i)$ are the nodes with arcs to the node x_i. In the case of an acyclic model, this score will be the joint probability of the event x, $P(x)$.

This model uses templates to extract features. These features include the following features: feature (t_0, w_0), features for capitalization and spelling of the word, left and right context features and various lexical features as described in Toutanova [40].

For unknown words, it uses a set of features from Ratnaparkhi [44], which include using character *n*-gram prefixes and suffixes (for *n* up to *4*), and other features such as capitalization, hyphens, and numbers. With so many features in the model, overtraining is a distinct possibility when using pure maximum likelihood estimation. This is avoided by using a Gaussian prior (also known as quadratic regularization or quadratic penalization) which avoids high feature weights unless they produce great score gain.

3.3. Guided learning for bidirectional sequence classification

Libin Shen and his collaborates proposed a perceptron-like guided learning, a new learning framework for bidirectional sequence classification that integrates classification of individual tokens and inference order selection into a single learning task [41].

Compared to other systems that use a perceptron-like algorithm, this model introduces a bidirectional search strategy. Instead of forcing the order of the tagging in a left-to-right fashion, any tagging order is allowed.

This approach uses scores for taking different actions and incorporates the easiest-first approach to learn the order of inference during the training phase. The pseudocode for guided learning algorithm proposed by Shen is shown below.

```
Algorithm for inference
Require: token sequence w1 … wn
Require: beam width B
Require: weight vector w

Initialize P, the set of accepted spans
Initalize Q, the queue of candidate spans
repeat
    span p′ ← argmax p∈Q U (p.S.T.A)
    update P with p′
    update Q with p′ and P
until (Q ≠ ∅)
```

It uses spans that hold different hypotheses about the possible sequence of tags. To reduce the search space, it uses beam width to store only the top scored hypothesis. Spans are started and grown by taking tagging actions.

Three kinds of actions are available:

- Start a new span by labeling a token which doesn't have any context
- Expand an existing span by labeling a token adjacent to the span
- Merge two spans by labeling the token between them.

In this last case, the two originating spans become subsequences of the resulting span, and the labeling action of the token between the spans use both right and left context information. The algorithm terminates when the whole token sequence is contained in one single span.

As a result of the freedom in the inference order, this approach can first tag the known words postponing the tagging of unknown words when a larger context is available.

3.4. Dynamic Feature Induction

The latest approach that was used to train a POS tagger was introduced in 2016 by Jinho Choi [30]. The technique is called dynamic feature induction. It keeps inducing high dimensional features automatically until the feature space becomes 'more' linearly separable.

There are many approaches that handle the problem of POS tagging using the linear kernel (for example Perceptron, Support Vector Machines) and have performed very well at this task. One of the reasons for this is the very high dimensional vector space such that it is rather forced to be linearly separable. However, if the feature space is not linearly separable, one could introduce 'higher' level features from 'lower' level features. As new features are introduced, the features' space becomes linearly separable at some point. Usually, finding good 'lower' level features is the task that has been explored well enough, many linguists have worked on that. But, finding 'higher' level of features could be an intensive manual work.

Thus, the dynamic feature induction is used to introduce new high dimensional features by joining together until the feature space becomes linearly separable (there is

no guarantee that this condition can be reached). Here is the broad overview of the process of dynamic features induction:

Given a training instance (x_1, y_1), where x_1 is a feature set and y_1 is the gold label, the classifier predicts the label.

Let us refer "strong features for y against y'" to features that give strong clues for distinguishing y from y'. If y'_1 is not equal to y_1, strong features for y_1 against y'_1 in x_1 are selected and combinations of these features are added to the induced feature set ***F*.** Given a new training instance (x_2, y_2), combinations of features in x_2 are checked by ***F*** and appended to x_2 if allowed:

- The extended feature set x_2 is fed into the classifier. If y'_2 is equal to y_2, no feature combination is induced from x_2.
- High dimensional features in **F** are learnt at the same time with low dimensional features. During decoding, the feature set is extended by the induced features in **F** and the new extended set is used to make the prediction.
- Theoretically, the size of the set **F** can grow up to $|\mathbf{X}|^2$ where $|\mathbf{X}|$ is the size of low features, but according to Choi it usually is a quarter of $|\mathbf{X}|$ in practice.

4. Results

To point out the correctness and efficiency of the approach, a brief introduction of some state-of-the art POS taggers is presented. Particular attention is paid on the languages which are part of the Multext-East initiative. Then, a presentation and discussion of the obtained results follows. First, a baseline POS tagger is presented. This tagger is actually a Suffix tagger and it performs exceptionally well compared to TnT. This shows that there could be significant improvement in TnT model if the corpus is bigger. Next, a comparison of the built models is presented and their performance on known and unknown words. At the end, an error analysis is presented.

4.1. Review of the state-of-the-art POS taggers

POS tagging is one of the first NLP activities. It was successfully done for most European languages. However, for some specific purposes, such as the POS tagging and dependency parsing, even human experts reach an accuracy of "modest" 96.95% [42]. These results are comparable with the automatic POS tagging done using the state-of-the-art techniques, such as the bidirectional long short-term memory for sequence tagging (bi-LSTM) [45]. The exhaustive multilingual research using the bi-LSTM has achieved an average accuracy of 96.50%, with the superior result for the Slavic languages [46]. The prerequisite to implement this approach were language corpora consisting of at least 60000 manually tagged tokens with 17 POS tags defined as part of the Universal Dependencies project [47]. The accuracy after the multi-task learning, in which the labels are predicted jointly for all the languages, the accuracy of the Slavic languages was the following: 96.27% for Croatian, 96.97% for Slovene, 97.94% for Polish, 98.02% for Czech, and the superior 98.23% for Bulgarian [46]. Unfortunately, Macedonian language is neither a part of the Universal Dependencies project, nor it has a 60K token tagged corpus, so this approach was not applicable at the moment. POS

tagging within Multext-East corpus was done a decade ago. The results of monolingual unsupervised POS tagging for English, Bulgarian, Slovene, and Serbian varied between 85.05% for Serbian up to 90.71% for English [48]. Romanian language was more exhaustively researched using various techniques, including TnT tagger [49]. The most successful result of 97.82% over Orwell's "1984" was obtained using the QTAG tagger [50]. Using the same corpus, error rates for Czech, Hungarian, English, Estonian, and Slovene were much higher. However, this is a rather old research, so its limited efficiency is expected. The best accuracy of 96.59% was achieved for Slovene [52].

The presentation and discussion of the obtained results follows. First, a baseline POS tagger is presented. This tagger is actually a Suffix tagger and it performs exceptionally well compared to TnT. This shows that there could be significant improvement in TnT model if the corpus is bigger. Next, a comparison of the built models is presented and their performance on known and unknown words. At the end, an error analysis is presented.

4.2. Baseline

First, a baseline tagger using the most-frequent tag as a starting point for the POS tagging process has been built. This tagger ignores the context, but assigns the tag that has previously been assigned to the word in the training set. It is important to notice that if there is a tie of the assigned tags in the training set to a particular word, then the tag is randomly chosen.

Even with this simple approach there are three options how unknown words should be tagged:

- Option 1: Consider them as wrong
- Option 2: Assign the overall most-frequent tag seen in the training set
- Option 3: Use a suffix tagger trained on the training set

In the training set, nouns were the most-frequent class, thus nouns are chosen in option 2 above. The results are presented in Table 3. They indicate that using unigrams and suffix tagger for unknown words, very high accuracy can be achieved. But this is very optimistic because the suffix tagger is trained on the same training set and it is very probable that this accuracy would be lower if it was tested on different test sets, which do not originate from the same novel. However, this is a good indication for further research.

Table 3. Most-frequent tag baseline taggers.

Baseline tagger	Accuracy
Unknown words are wrong	86.25%
Unknown words are nouns	90.15%
Suffix tagger for unknown words	94.01%

4.3. Comparative analysis of the obtained results

The dataset, which was introduced in the Section 2.1 was randomly divided into training and testing set using 70%-30% split.

The same training and test datasets were used for all models that were built (Table 4).

Table 4. Training and testing sets.

Dataset	Sentences	Tokens
Training	4667	76364
Testing	2000	32446
Known tokens	58544	29316
Unknown tokens	33510	3130

For TnT model, two separate experiments were done. First, without any extra model that would tag unknown words, then a suffix tagger of length 3 was built on the training set and was used to tag unknown words.

The suffix tagger that was trained, not only helps in tagging unknown words, but it also increases the accuracy for known words. This is due to the fact that preceding decisions have an impact on the following decisions.

The model that used the cyclic dependency network creates features for the current word and the preceding and following word. Features of the preceding and following tag of the current word, a combination of the current word and the previous tag, similarly a feature with the tag of the next word is also used. It also uses prefixes and suffixes of the current word of length up to 10 to tag unknown words.

The model that was built based on bidirectional guided learning framework used the features which are shown in Table 5 where *w* denotes a word and *l* denotes the already assigned label (class). The model was tested with the beam sizes 1, 2 and 3 and the model with beam size 3 performed the best on the testing set.

Table 5. Context feature templates for bidirectional guided learning framework

Word features	$[w_{-2}],[w_{-1}],[w_0],[w_1],[w_2],[w_{-1,},w_0],[w_0,w_1]$
Left context features	$[l_{-2}]$, $[l_{-1}]$, $[l_{-2},l_{-1,}]$, $[l_{-2},w_0]$, $[l_{-2},w_0],[l_{-2},l_{-1},w_0]$
Right context features	$[l_1]$, $[l_2]$, $[l_1,l_2]$, $[l_1,w_0]$, $[l_2,w_0]$, $[l_1,l_2,w_0]$
Bidirectional features	$[l_{-1},l_1]$, $[l_{-1},l_1,w_0]$

As Table 6 suggests, the most important part for the performance of the tagger was the performance on unknown words. The testing set contains 3130 unknown tokens or around 11% of the tokens in the test set. This is an indicator that if the training set is larger, better performance of the taggers could be expected.

The results in Table 6 are an improvement to the last obtained results [37] because of the revision and correction of some mistakes that were made during the initial process of manual tagging of the corpus.

Table 6. Compared accuracy of all the four examined models

Model	Sentences	Tokens
TnT	21.17%	86.46%
Known words		95.55%
Unknown words		1.27%
Ambiguous words		88.07%
Non-ambiguous words		85.74%
TnT + Suffix for unknown words	46.90%	94.22%
Known words		96.35%
Unknown words		74.22%
Ambiguous words		89.91%
Non-ambiguous words		96.14%
Cyclic dependency network	70.85%	97.50%
Known words		97.91%
Unknown words		92.36%
Ambiguous words		94.21%
Non-ambiguous words		98.96%
Bidirectional guided learning	70.05%	97.43%
Known words		97.87%
Unknown words		93.22%
Ambiguous words		94.19%
Non-ambiguous words		98.87%
Dynamic feature induction	60.20%	95.18%
Known words		95.70%
Unknown words		90.32%
Ambiguous words		91.33%
Non-ambiguous words		96.90%

4.4. Error analysis

Apart from estimating the accuracy of the POS models, it was very important to judge what type of mistakes the automatic taggers made, bearing in mind that even human experts sometimes can't agree upon the exact POS tag of a particular word in some context. Thus, the examination of the errors is very beneficial for further improvements of the tagger. The whole confusion matrix included all the word classes, and it was already presented in the ACM paper [37].

Not every mistake that the tagger made had an equal weight. Some mistakes may introduce problems in the prediction of other tags in the sentence. Wrong tagged words can cause even more damage when the POS tagging is used to preprocess a text for dependency parsing. As it was mentioned in Section 2.2, there are some very common ambiguities in Macedonian language. Here are the results how the tagger performed on those words in the testing set.

The ambiguities mentioned in the section 2.2 were expectedly very frequent. Namely, 85 out of 163 wrongly recognized adjectives (52.14%) were assigned as adverbs. Conversely, 88 out of 176 missed adverbs were actually adjectives. However,

such amount of misclassified adjectives and adverbs is below 1.6% for the adjectives and 2.3% for the adverbs (see Table 2 for more details). The most frequently confused words were: *лесно (lesno / easy, easily)*, *добро (dobro / good, well)*, *вистински (vistinski / real, really)*, *објективно (objektivno / objective, objectively)*, *надворешно (nadvoreshno / outside)*, *изобилно (izobilno / abundant, abundantly)*, *неконтролирано (nekontrolirano / uncontrolled)*, *заеднички (zaednichki / joint. jointly)*, and *сериозно (seiozno / serous, seriously)*.

Nouns were mixed with the verbs 19 out of 84 times, while verbs were mixed with nouns 20 out of 67 times. The examples include the following words: *врати (vrati / doors, to return), работи (things, to work), говори (speeches, to speak), правите (pravite / lines, to make), опомена (opomena / warning, to remind), трага (traga / track, to search), проби (probi / examples, to break through), мислите (mislite / thoughts, to think), промени (changes, to change)*, and *мува (muva / fly, to roam)*.

Considerable amount of wrong classification appeared between the conjunctions and particles, which were manually easily resolved according to the context they appear. The mistakes when the proper noun class was misidentified with the noun class was due to the fact that there were significantly less proper nouns and they were usually surrounded by a similar context as common nouns.

The words with several POS tags were not so common, may be because they are not frequent in the corpus. The word *само* (*samo / only*) was misclassified 23 times. Even 19 of its occurrences as a particle, it was classified as an adverbs. The word *едно* (*edno / one*) was considered 3 times as a number, although it was an adjective. In total, only 12 mistakes were made with it. Finally, *што (shto / what, that*) was incorrectly assigned 53 times, 26 times it was a conjunction, and 14 times an adverb, both classified as pronouns.

4.5. Comparison to other languages

Orwell's novel '1984' as part of the aligned multi-corpus in MULTEXT-East project, has been a subject of research in many other languages that are part of the foregoing project.

As it was shown in the results, the overall performance of the tagger is relying on the accuracy of unknown words. In addition, very interesting comparisons would be the performance of the tagger on words that have ambiguity. However, very few of the authors have reported these types of results in their papers.

Table 7. Compared accuracy of POS tagging in different languages on Orwell's '1984'

Language	Accuracy	Ambiguity	Accuracy on unknown	Acc. on ambiguous
Macedonian (current)	97.50%	36%	92.36%	94.21%
Macedonian [36]	88.00%	39%	91.00%	87.00%
Slovene [52]	96.59%	38%	85.30%	/
Romanian [50]	97.82%	40%	/	/
Czech [51]	92.96%	46%	/	/
English [51]	97.57%	39%	/	/

Accordingly, we show a comparison in Table 7 of the overall accuracy, the percentage of ambiguous tokens in the corpus and the performance on the tagger on unknown and ambiguous words (when results are available).

5. Conclusion and further work

This paper presented the whole process of creating a POS tagger of Macedonian language. Unlike all the previous attempts that have been done in the area of POS tagging and annotation of Macedonian language, this work used a completely manually annotated and purified corpus. Therefore, the results are more reliable and currently, they are the best that have been reported. The contributions of this work are multiple.

First, a completely tagged corpus, of Orwell's novel "1984" was created for Macedonian language by merging the results of manual annotation by two different annotators. Next, a comparison of different machine learning models was presented. By using different models, the best achieved accuracy was 97.50% using the cyclic dependency network.

The results have shown that by having larger corpora, the tagger would be more robust. By doing a review of the manual tagged corpora, an improvement of 1% has been achieved compared to the previous results [37] on the same dataset and with the same methods. However, they have also revealed that with the larger corpora, one should not expect an improved accuracy higher than 1%. However, the inclusion of professional linguists willing to manually disambiguate the ambiguities would certainly improve the quality of the tagger.

The results reported in this paper are much higher than previously reported results for Macedonian language that took different approaches, for example in Aepli et al. research [36]. Also, the first POS tagger of Macedonian language is publicly available. It can be found and examined at http://bonchanoski.com/postagger/en/tag.

The future work would have to include full MSD annotation and include larger corpora. In addition, it is intended to pay more attention to long-distance dependencies that could reduce the errors that the tagger currently makes when, for instance, it tags the adjectives and the adverbs with an identical form. Also, it would be of crucial interest to obtain results of the performances of the tagger on corpus that is different from the training corpus.

Acknowledgements

This work was partially financed by the Faculty of Computer Science and Engineering at the Ss. Cyril and Methodius University.

References

1. Leech, G., A. Wilson.: Recommendations for the Morphosyntactic Annotation of Corpora EAGLES Report. (1996).

2. Chowdhury, G. G.: Natural language processing. Annual review of information science and technology 37, no. 1 51-89. (2003)
3. Ide, N., Romary, L.: International standard for a linguistic annotation framework. Natural language engineering 10, no. 3-4, 211-225. (2004)
4. ISO/TC 37: Language and terminology (1947) [Online]. Available: https://www.iso.org/committee/48104/x/catalogue/p/0/u/1/w/0/d/0 (current February 2018)
5. Marcus, M. P., Marcinkiewicz, M. A., Santorini B.: Building a large annotated corpus of English: The Penn Treebank., Computational linguistics 19, no. 2, 313-330. (1993)
6. Francis, W. N.: A Standard Sample of Present-Day English for Use With Digital Computers. (1964)
7. Taylor, A., Marcus M., Santorini B.: The Penn treebank: an overview.. Treebanks, Springer, Dordrecht, 5-22. (2003)
8. Xue, N., Xia, F., Chiou, F.D. Palmer, M.: The Penn Chinese TreeBank: Phrase structure annotation of a large corpus. Natural language engineering, 11(2), 207-238. (2005)
9. Maamouri, M., Bies, A., Buckwalter, T., Mekki, W.: The penn arabic treebank: Building a large-scale annotated arabic corpus. In NEMLAR conference on Arabic language resources and tools (Vol. 27, pp. 466-467). (2004)
10. De Marcken, C. G.: Parsing the LOB corpus. In Proceedings of the 28th annual meeting on Association for Computational Linguistics, 243-251. Association for Computational Linguistics, (1990)
11. Marshall, I.: Choice of grammatical word-class without global syntactic analysis: tagging words in the LOB corpus. Computers and the Humanities, 17(3), 139-150, 1983.
12. Silberztein, M.: NooJ: a linguistic annotation system for corpus processing. In Proceedings of HLT/EMNLP on Interactive Demonstrations. 10-11. Association for Computational Linguistics. (2005)
13. Droganova, Kira. "Building a Dependency Parsing Model for Russian with MaltParser and MyStem Tagset." In *International Workshop on Treebanks and Linguistic Theories (TLT14)*, p. 268. 2015.
14. Nivre, J., Hall, J., Nilsson, J., Chanev, A., Eryigit, G., Kübler, S., Marinov, S. Marsi, E.: MaltParser: A language-independent system for data-driven dependency parsing. Natural Language Engineering, 13(2), 95-135. (2007)
15. Hajič, J., Hajičová, E., Mikulová, M. Mírovský, J.: Prague Dependency Treebank. In Handbook of Linguistic Annotation, 555-594, Springer, Dordrecht. (2017)
16. Hajič, J., Smrz, O., Zemánek, P., Šnaidauf, J. and Beška, E.: Prague Arabic dependency treebank: Development in data and tools. In Proc. of the NEMLAR Intern. Conf. on Arabic Language Resources and Tools, 110-117. (2004)
17. Aduriz, I.: Construction of a Basque dependency treebank. (2003)
18. Głowinska, K., Przepiórkowski, A.: The design of syntactic annotation levels in the National Corpus of Polish. In Proceedings of LREC 2010. (2010)
19. Przepiórkowski, Adam, Rafal L. Górski, Marek Lazinski, and Piotr Pezik. "Recent Developments in the National Corpus of Polish." In *LREC*. 2010.
20. Dimitrova, L., Ide, N., Petkevic, V., Erjavec, T., Kaalep, H. J., Tufis, D.: Multext-east: Parallel and comparable corpora and lexicons for six central and eastern European languages. In Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics-Volume 1, 315-319. Association for Computational Linguistics. (1998)
21. Erjavec, T.: MULTEXT-East. In *Handbook of Linguistic Annotation* (pp. 441-462). Springer, Dordrecht. (2017)
22. Erjavec, T: The Multext-east Slovene lexicon. In Proceedings of the 7th Electrotechnical Conference ERK, Volume B. 189-192. 1998
23. Tadić, M.: Building the Croatian national corpus. In Third International Conference on Language Resources and Evaluation LREC2002. ELRA. (2002)

24. Krstev, C., Vitas, D. Erjavec, T.: MULTEXT-East resources for Serbian. In Zbornik 7. mednarodne multikonference Informacijska druzba IS 2004 Jezikovne tehnologije 9-15 October 2004, Ljubljana, Slovenija, 2004. Erjavec, Tomaž and Zganec Gros, Jerneja. (2004)
25. Ivanovska, A., Zdravkova, K., Erjavec, T., Džeroski, S.: Learning rules for morphological analysis and synthesis of Macedonian nouns, adjectives and verbs, In Proceedings of 5th Slovenian and 1st international Language Technologies Conference, Jozef Stefan Institute, Ljubljana, 140-145. (2006)
26. Koneski, K., Velkovska, S., Cvetkovski, Zh. (eds.) Interpretative dictionary of Macedonian language Vol 1 – VI, Institute of Macedonian Language "Krste Misirkov". (2003 - 2014)
27. Erjavec, T., Derzhanski, I., Divjak, D., Feldman, A., Kopotev, M., Kotsyba, N., Krstev, C., Petrovski, A., QasemiZadeh, B., Radziszewski, A., Sharoff, S.: MULTEXT-East non-commercial lexicons 4.0. (2010)
28. Derebej, S.: Digital dictionary of Macedonian language, version 1.2.002 (2018) [Online]. Available: https://makedonski.info/ (current February 2018)
29. Bonchanoski, M., Zdravkova, K.: Machine Learning-based approach to automatic POS tagging of Macedonian language. In *Proceedings of the 8th Balkan Conference in Informatics* (p. 11). ACM. (2017)
30. Choi, J. D.: Dynamic feature induction: The last gist to the state-of-the-art. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 271-281. (2016)
31. Lawrence, P.N., Loafman, K.W. and Benno, S.A., Krona Acquisitions Corp and Syngence Corp, System and method for identifying relationships between database records. U.S. Patent 7,246,129. (2007)
32. Erjavec, T.: MULTEXT-East: morphosyntactic resources for Central and Eastern European languages. Language resources and evaluation, 46(1), 131-142. (2012)
33. Manandhar, S., Džeroski, S.. Erjavec, T.: Learning multilingual morphology with CLOG. In Procedings of International Conference on Inductive Logic Programming, 135-144. Springer, Berlin, Heidelberg. (1998)
34. Vojnovski, V., Džeroski, S. Erjavec, T : Learning PoS tagging from a tagged Macedonian text corpus. In Proceedings of SIKDD 2005 (2005).
35. Erjavec, T., Sárossy, B.: Morphosyntactic tagging of Slovene legal language. *Informatica*, *30*(4) (2006)
36. Aepli, N., von Waldenfels, R., Samardžić, T.: Part-of-speech tag disambiguation by cross-linguistic majority vote. In *Proceedings of the First Workshop on Applying NLP Tools to Similar Languages, Varieties and Dialects*, 76-84. (2014)
37. Bonchanoski, M. and Zdravkova, K., 2017. Automatic POS tagging of Macedonian Language. [Online]. Available: http://eprints.finki.ukim.mk/11391/1/978-608-4699-07-1_pp136-140.pdf, (current February 2018)
38. Martin, James H., and Daniel Jurafsky. Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition. Pearson/Prentice Hall, 2009.
39. Brants, T.: TnT: a statistical part-of-speech tagger. In *Proceedings of the sixth conference on applied natural language processing*, pp. 224-231. Association for Computational Linguistics. (2000)
40. Toutanova, K, Dan Klein, Christopher D. Manning, and Yoram Singer. Feature-rich part-of-speech tagging with a cyclic dependency network." In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pp. 173-180. Association for Computational Linguistics, (2003)
41. Shen, Libin, Giorgio Satta, and Aravind Joshi. "Guided learning for bidirectional sequence classification." In *Proceedings of the 45th annual meeting of the association of computational linguistics*, pp. 760-767. (2007)

42. Mukherjee, A., Kübler, S., & Scheutz, M. (2017). Creating POS tagging and dependency parsing experts via topic modeling. In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers (Vol. 1, pp. 347-355).
43. Forney, G. D.: "The Viterbi algorithm." Proceedings of the IEEE 61, no. 3 (1973): 268-278.
44. Ratnaparkhi, A.: A maximum entropy model for part-of-speech tagging. In Proceedings of the conference on empirical methods in natural language processing. Vol. 1, 133-142. (1996)
45. Huang, Z., Xu, W., & Yu, K. (2015). Bidirectional LSTM-CRF models for sequence tagging. arXiv preprint arXiv:1508.01991.
46. Plank, B., Søgaard, A., & Goldberg, Y. (2016). Multilingual part-of-speech tagging with bidirectional long short-term memory models and auxiliary loss. *arXiv preprint arXiv:1604.05529*.
47. Nivre, Joakim, Marie-Catherine De Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajic, Christopher D. Manning, Ryan T. McDonald et al. "Universal Dependencies v1: A Multilingual Treebank Collection." In *LREC*. 2016.
48. Snyder, B., Naseem, T., Eisenstein, J., & Barzilay, R. (2008, October). Unsupervised multilingual learning for POS tagging. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing* (pp. 1041-1050). Association for Computational Linguistics.
49. Cristea, D., & Forăscu, C. (2006). Linguistic resources and technologies for Romanian language. *Computer Science Journal of Moldova*, *14*(1), 40.
50. Tufis, D., & Mason, O. (1998, May). Tagging Romanian texts: a case study for qtag, a language independent probabilistic tagger. In *Proceedings of the First International Conference on Language Resources and Evaluation (LREC)* (Vol. 1, pp. 589-596).
51. Hajič, J. (2000, April). Morphological tagging: Data vs. dictionaries. In *Proceedings of the 1st North American chapter of the Association for Computational Linguistics conference* (pp. 94-101). Association for Computational Linguistics.
52. Dzeroski, S., Erjavec, T., & Zavrel, J. (2000). Morphosyntactic Tagging of Slovene: Evaluating Taggers and Tagsets. In LREC.

Martin Bonchanoski has got his Bachelor degree in Computer Science at Faculty of Computer Science and Engineering at university 'Ss. Cyril and Methodius' in Skopje, Republic of Macedonia (2016). He is a young researcher who has published 3 papers at international conferences. His research interest is mainly focused on Natural Language Processing with special interest in Macedonian language.

Prof. Katerina Zdravkova has been working in the area of natural language processing for more than 30 years. Her major interests involve morphological segmentation, annotation and multi-word expressions, predominantly for Macedonian language. Apart from NLP, her research interests also include computer ethics and technology enhanced learning.

Received: March 10, 2018; Accepted: September 3, 2018