

Combining POS Tagging, Dependency Parsing and Co-referential Resolution for Bulgarian

Valentin Zhikov and Georgi Georgiev

Ototext AD
Sofia, Bulgaria
valentin.zhikov@ototext.com
georgiev@ototext.com

Kiril Simov and Petya Osenova

Linguistic Modelling Department
IICT-BAS, Sofia, Bulgaria
kivs@bultreebank.org
petya@bultreebank.org

Abstract

This paper proposes a combined model for POS tagging, dependency parsing and co-reference resolution for Bulgarian — a pro-drop Slavic language with rich morphosyntax. We formulate an extension of the MSTParser algorithm that allows the simultaneous handling of the three tasks in a way that makes it possible for each task to benefit from the information available to the others, and conduct a set of experiments against a treebank of the Bulgarian language. The results indicate that the proposed joint model achieves state-of-the-art performance for POS tagging task, and outperforms the current pipeline solution.

1 Introduction

Advanced language technology applications depend on various forms of preprocessing, such as POS tagging, parsing, co-reference resolution, word sense disambiguation, etc. Although in ideal settings these tasks have satisfactory solutions on their own, their combination in a pipeline is related to a significant decrease in accuracy at each consequent stage of analysis. Recently, models that enable the single-step handling of multiple tasks have gained popularity, as they improve on the performance achieved by pipeline approaches. They take advantage of the interaction among the various levels of linguistic knowledge. Here we propose a model that challenges three tasks simultaneously: POS tagging, dependency parsing and co-reference resolution (within a sentence). The experiments are performed on data from the Bulgarian HPSG-based treebank — BulTreeBank. Our motivation to attempt solving these particular problems via a single model is many-fold: (1) avoiding the accumulation of errors inherent to pipeline processing, (2) overcoming the low speed

of model-chaining approaches, (3) confirming the success of previous developments in joint modeling; and last but not least, (4) assessing the benefits of modeling the interactions that exist among morphology, syntax and discourse.

Pipeline approaches follow a sequence of processing that reflects the traditional levels of analysis within linguistics: syntax depends on morphology; co-reference resolution depends on morphology and syntax. Thus, dependency arcs are determined by the grammatical features of the wordforms; co-reference chains depend on the grammatical features of wordforms and the configuration of the dependency arcs. Unfortunately, this style of processing does not necessarily lead to optimal results. One should keep in mind that some alternative interaction paths and interdependencies exist among the linguistic levels, and this interdependence can be accounted for in order to achieve a better solution for each task. Two phenomena in Bulgarian that illustrate this statement are: (1) co-reference links between dative verbal clitics and nouns (within a prepositional phrase, expressing the indirect object of the same verb) have common number and gender features; (2) unexpressed subjects participate in co-reference chains of control, binding, etc. constructions. We propose a model capable of handling such interactions among the different linguistic levels. We define an extended dependency tree that incorporates service nodes and links, through which additional knowledge, such as POS tag candidates, correct POS tags and co-reference relations, can be fed into the MSTParser algorithm for non-projective dependency parsing (McDonald et al., 2005). The sentences in the treebank are projected as extended dependency trees, and the parser is applied to their new representation. Although the proposed model addresses the Bulgarian language, it is also applicable to other languages, provided that all necessary resources are available.

The structure of this paper is as follows: in Section 2, we introduce related work; in Section 3, we discuss the relevant annotations available in the Bulgarian treebank; Section 4 presents the proposed approach for joint modeling, Section 5 elaborates on our experimental settings and the obtained results; Section 6 concludes the paper.

2 Related work

We are not aware of other studies that propose joint models for Bulgarian, and to the best of our knowledge, attempts at combining the three tasks (POS tagging, dependency parsing and co-reference resolution) in a joint model have not been described in the literature either.

Our approach is inspired by works such as (Finkel and Manning, 2010), (Bohnet and Nivre, 2012) and (Qian and Liu, 2012). Finkel and Manning (2010) report on combining NER and parsing tasks in a joint model. One similarity with our task is the understanding that the separate tasks can help each other in various not-always-subsequent executions. Another one is the fact that the explored algorithm is extended. The difference is that the authors rely on a feature-rich CRF parser, while our algorithm is based on an online large-margin learning algorithm.

Bohnet and Nivre (2012) studies the combination of two tasks (POS tagging and Dependency labeled non-projective parsing) against datasets in four languages, and the reported results indicate an improvement over the pipeline-generated output for all considered languages. The algorithm behind their architecture is transition-based.

The reported results indicate that combining POS tagging and dependency parsing could be a successful step not only for morphologically rich languages (such as Czech and German), but also for languages where POS ambiguities are abundant (such as Chinese). This work illustrates the superiority of joint models in settings rather similar to our own. The authors added features for improving the POS tagging task within the combined model. We also followed this strategy.

Our work differs in the choice of an algorithm (Maximum Spanning Tree Model), and in the greater number of problems tackled by the proposed model. The motivation for choosing the approach of the MSTParser is that two of the tasks that we handle can be non-local, and the algorithm may require information from distant nodes in or-

der to find an appropriate solution. Therefore, a straight adaptation of the transition-based model is not possible.

Qian and Liu (2012) focuses on the modelling of three tasks for Chinese - word segmentation, POS tagging and parsing. The models for each task are trained separately, while the unification of predictions is performed during the decoding phase. As in the previous paper, the authors report improvements over the pipeline results for Chinese. The similarity is that our approach also considers three tasks in one model for one language with a modified algorithm.

Our approach differs in the following aspects: the third task is not identical. In our case it is the addition of co-reference chains instead of the specific for Chinese word segmentation module. Bulgarian is a morphologically rich language in comparison to Chinese - hence, the POS tagging model is more complex. The parsing task uses dependencies instead of the CFGs used in the case of the Chinese parser. Our model does not train the tasks separately, with specific models, before combining them, and the joint model is used during the development and exploitation of the proposed parser. Our aim is to combine 3 closely related tasks, which have not been addressed widely in NLP, and to evaluate their impact on the processing of Bulgarian. The complexity of the joint task is high not only due to the number of modules incorporated in the model, but also to the morphosyntactical richness of the language addressed in our work.

Below we describe our dataset, before we continue discussing the algorithm that handles the joint modeling task.

3 The Linguistic Annotation of the Bulgarian Treebank (BulTreeBank)

BulTreeBank provides rich linguistic information that goes beyond syntactic annotation. It comprises the full grammatical tags, lemmas for all wordforms, syntactic relations (HPSG), named entities, as well as co-references within each sentence. Since parts of speech, syntactic and co-reference relations have been incorporated in our joint modeling effort, we will outline the specifics of their annotation within the dataset.

As we have already mentioned, Bulgarian is a morphologically rich language. Morphological richness has many varieties from a typological

point of view. Bulgarian has a very rich verb system, and it is an inflective language, whose complete part of speech tagset comprises about 680 tags¹. As this circumstance causes sparseness and increases the modeling complexity, we opt in for filtering the input with the aid of a rich morphological lexicon and morphological guessers. Besides the original HPSG-based corpus, there is a dependency version of BulTreeBank, derived from the original dataset. More details regarding the types of dependency relations available in it are enlisted at <http://www.bultreebank.org/dpbtb/>.

In Figure 1, an HPSG-based tree of the sentence "Vednaga odobri namerenieto na sestra si" ('Immediately approved intention of sister his', He approved his sister's intention immediately) is shown. This example illustrates the way in which the HPSG-based version of the dataset encodes dependency information (the "NPA" tag stands for nominal phrases of type head-adjunct). Another noteworthy detail is the co-reference link between the un-expressed subject and the reflexive possessive pronoun. In the HPSG-based version of the treebank, the unexpressed subject is represented explicitly only in cases when it participates in a co-reference chain, as shown in the sample sentence. It is considered to be a property of the verb node, and not part of the constituent structure.

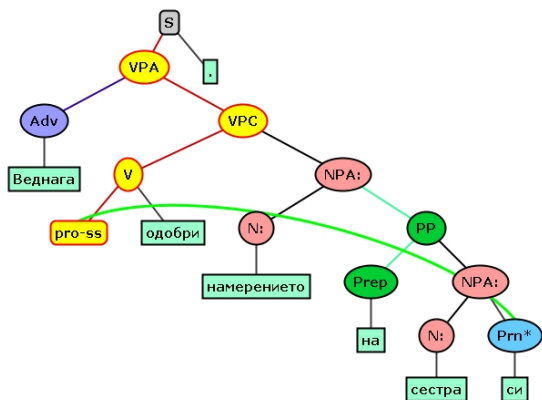


Figure 1. HPSG-based tree.

Figure 2 provides a view on the same sentence after its conversion to dependency format. The head-adjunct relation found within the lowest NPA in the tree has been projected into a head-modifier relation. Co-reference arcs have not been transferred into the dependency version of the treebank used within the CoNLL 2006 shared task. We have

¹<http://www.bultreebank.org/TechRep/BTB-TR03.pdf>

added them specially for the modeling effort reported in this paper. Here, co-references are represented as secondary edges connecting the word nodes, and arc labels are represented as ovals situated between the connected word pairs.

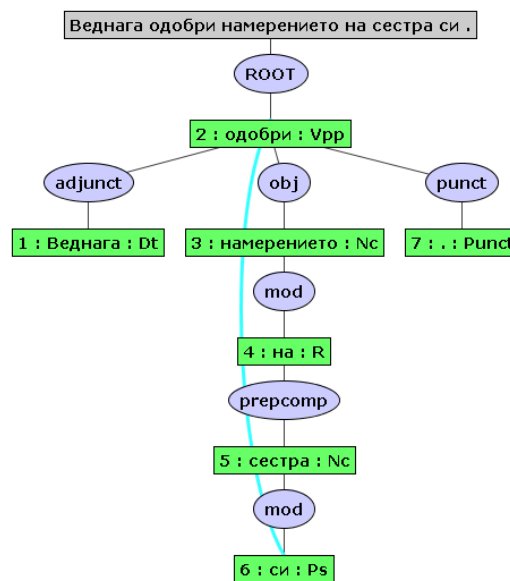


Figure 2. Dependency tree.

The annotation of BulTreeBank complies with the definition of co-reference resolution as the identification of expressions that reference a common discourse entity (Recasens et al., 2010). From a semantic perspective, co-references include three types of relations: "equality", "member-of" and "subset-of". Reflected linguistic phenomena include: pro-dropness (when co-referentially bound), subject and object control, secondary predication, binding, and nominalizations. Co-references are found in the following set of dependency relations: coordination, subordination, complementation, adjunction and modification. The annotated co-reference chains within the treebank amount to 5,312. On average every third sentence contains at least one co-reference chain. Thus, the impact of the co-references within Bulgarian grammar is clearly indicated.

4 Maximum Spanning Tree Model of the Joint Task

4.1 Extended dependency tree model

In this section we introduce a method for incorporating part-of-speech and co-reference tags into the tree-representation of a sentence. This transformation enables the direct application of the maximum spanning tree non-projective parser developed by McDonald et al. (2005). We define the

#	System	POS		Co-reference		Dependency		
		Accuracy (%)	Prec (%)	Recall (%)	F	LAS (%)	UAS (%)	LA (%)
1	features&morph	95.99	80.90	33.08	46.96	81.22	85.12	88.96
2	features&decomp. morph*	95.52	81.04	32.08	45.96	80.50	84.55	88.59
3	1&word context	95.95	80.97	33.23	47.12	81.42	85.35	88.95
4	3&distances	95.98	82.03	37.06	51.05	81.82	85.70	89.32
5	4&context-bigrams	97.12	81.77	35.38	49.39	82.29	86.19	89.65
6	5&additional conjunctions	97.13	81.16	34.30	48.22	82.39	86.17	89.64

Table 1: Evaluation results on the test dataset.

Labeled Arc Score (LAS): Accuracy computed over both correctly connected and properly labeled arcs.

Unlabeled Arc Score (UAS): Accuracy computed over correctly connected arcs.

Label Accuracy (LA): Accuracy computed over correctly labeled arcs.

Prec(ision), Recall, F: Correspond to the standard F1 metric and its components.

analysis of a sentence as a tree that includes some new types of service nodes in addition to the nodes that represent words. Service nodes connect to either words or other service nodes, in accordance with a set of rules that we describe in detail in 4.2.

Let us have a set G of POS tags, and a set D of dependency tags ($ROOT \in D$). Let us have a sentence $x = w_1, \dots, w_n$. A *tagged dependency graph with co-reference relations* is a directed tree $T = (V, A, \pi, \delta, C)$ where:

1. $V = \{0, 1, \dots, n\}$ is an ordered set of nodes, that corresponds to an enumeration of the words in the sentence (the root of the tree has index 0);
2. $A \subseteq V \times V$ is a set of arcs;
3. $\pi : V \rightarrow G$ is a partial labeling function from nodes to POS tags;
4. $\delta : A \rightarrow D$ is a labeling function for arcs;
5. 0 is the root of the tree
6. $C \subseteq V \setminus \{0\} \times V \setminus \{0\}$ is a set of undirected arcs representing the co-reference equality relation over the nodes of the dependency tree;

We will hereafter refer to this structure as a parse graph for the sentence x . Figure 2 illustrates one such parse graph.

As a first step of extending the tree, we assume a range of possible POS tags for each wordform in the sentence. Such a range of tags has to contain the correct tag for the wordform in the given context. The straightforward solution of assigning all the tags available in the tagset to each wordform makes the subsequent task of obtaining the correct tag infeasible, due to the great number of tags

available in BulTreeBank. In order to deal with this issue, we incorporate an inflectional lexicon (including a substantial set of entity names), which provides all possible tags for the wordforms available in it. Furthermore, we enable the handling of unknown words by applying a morphological guesser that suggests up to ten possible tags per wordform. Thus, we use the described components to yield a highly accurate and compact set of candidate POS tags.

These tags are included in the tree as service nodes. In the linear representation of the sentence, they are inserted after the node for the corresponding wordform, and before the node for the next wordform to the right. They are connected to the corresponding wordform with a special link \$TAG.

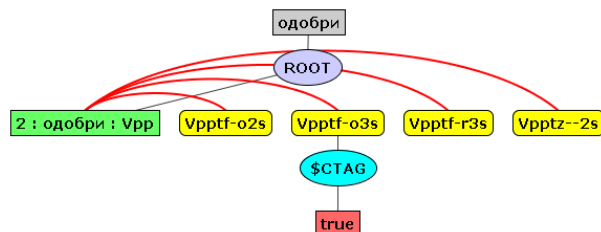


Figure 3. Subtree of the candidate POS tags and the correct tag for one word.

In order to indicate the correct tag, we introduce another type of service node. In the linear representation of the sentence, it is inserted after the last POS tag candidate node, and before the one corresponding to the next wordform to the right. This node is connected to the correct tag via a special arc \$TAG (correct tag). In this way, all information about the potential tags and the correct tag is represented in the form of a subtree, attached to the wordform. Figure 3 depicts the encoding of a word with POS tag ambiguity. The correct tag is

indicated: verb, personal, perfective, transitive, finite, aorist, third person, singular. The \$TAG arcs are represented as red links without labels. The \$CTAG arc is represented as an oval.

The next problem is representing the co-referential relations via the dependency tree. In order to do this, we introduce yet another type of service node, denoted as \$CR. Such nodes are inserted on the right side of the corresponding wordform node, and to the left of the first POS-tag candidate node in the linear representation of the sentence. We classify these nodes into two groups. The first group consists of nodes, attached to wordforms that do not participate in a co-reference relation with another wordform that precedes them in the sentence. These \$CR nodes are linked to their wordform with an arc labeled \$DI (discourse index), which might be linked to an entity in the discourse.

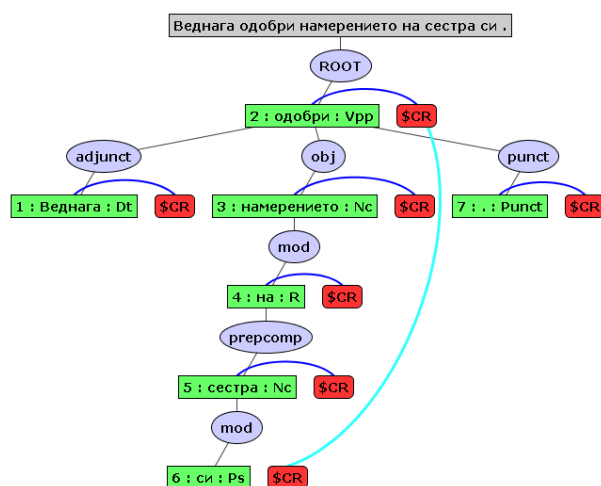


Figure 4. Representation of co-references as tree fragments.

The second group of \$CR nodes are those participating in a co-reference relation between their corresponding wordform and another wordform that precedes them in the sentence. We say that such nodes share a discourse index with a word preceding them in the sentence, and assign the \$SDI label to the arcs that interconnect such pairs service nodes. The nodes in the second group are not connected to their corresponding wordform nodes, but are instead connected to the co-reference nodes of the referenced entities. This approach allows us to represent the co-reference relations as supplementary tree fragments, attached to the original tree. Figure 4 presents an example of a sentence tree that contains both kinds of co-

reference nodes and the means through which they are connected to the graph.

\$DI arcs are depicted as dark blue links. In cases where a word participates in a co-reference chain with a word that precedes it, there is no link between the word and its \$CR node. Instead, its \$CR node is connected to the \$CR node of the first word in the co-reference chain. Such arcs (\$SDI) are depicted as light blue links.

Applying the described transformations allows us to obtain a tree representation of a tagged dependency graph that includes co-reference relations.

4.2 Constraining the edge generation mechanism of the MSTParser

Inference on the complex graph structures output by the proposed sentence representation technique leads to a significant computational overhead, given the specifics of the original MST algorithm that generates edges among all nodes in the graph. In order to reduce the feature space and simplify the learning task, we take advantage of the circumstance that service node connectivity is subject to a set of rules, and we eliminate edges that do not conform to these rules by modifying the feature-generation mechanism of the MST-Parser.

To this end, we assign empty feature vectors to the dependency arcs that do not comply with either of the following preconditions: (i) root nodes can only be linked to word nodes; (ii) word nodes can only be linked to their corresponding co-reference (\$CR) and POS candidate (\$TAG) nodes, other word nodes, or root nodes; (iii) co-reference (\$CR) nodes can only be linked to their corresponding wordform nodes (via a \$DI arc), or to the co-reference nodes of wordforms preceding them in the sentence (via a \$SDI arc); (iv) POS-candidate (\$TAG) nodes can be linked to their corresponding word node, and to the node that denotes the true POS tag (\$CTAG); and (v) node denoting true POS tags (\$CTAG) can only be linked to one of the POS candidate nodes (\$TAG) of the corresponding wordform.

Additionally, we introduce a set of linguistic rules to further reduce the number of co-reference arcs. Given a list of parts of speech that cannot take part in co-reference relations, and the set of candidate POS tags for a pair of wordforms, we assign empty feature vectors to the edges that

connect the co-reference nodes of the two words, when it is clear that they cannot be involved in a co-reference relation. In order to do so, we inspect the candidate tag set of each node, and check whether all of its candidate tags belong to either of the following part-of-speech classes (regular expressions that cover all tag variations available within BulTreeBank are provided in the brackets that follow the names of the individual classes): (1) particles ("T.*"); (2) adverbs ("D.*"); (3) interjections ("I"); (4) prepositions ("R"); (5) impersonal verbs ("Vn.*"); (6) conjunctions ("C.*"); (7) punctuation ("punct"); (8) gerunds ("V.*g").

4.3 Features incorporated in the joint model

In this section, we outline the set of features available to the algorithm during our joint modeling effort. Feature vectors are extracted on a per-edge basis, by applying a common set of rules over each pair of nodes that remains after the filtering step described earlier.

We use a feature naming convention that allows the classifier to discern six groups of features on the basis of the types of the interconnected nodes, i.e. different weights are learned for edges that connect different types of nodes. In this way, our model is aware of sets of features that correspond to the following dependency arc types: (i) word \rightarrow sentence root; (ii) word \rightarrow word; (iii) co-reference \rightarrow word (\$SDI); (iv) co-reference \rightarrow co-reference (\$DI); (v) POS candidate \rightarrow word; (vi) correct POS node \rightarrow candidate POS node. Furthermore, the features reflect the individual characteristics of the head and dependent nodes in each of these types of pairs. We provide details regarding each subset of features below.

Attachment distance is computed for each pair of interconnected nodes. Our algorithm provides two alternative modes for calculating the attachment distance - one that accounts for the presence of service nodes among the words, and one that ignores such nodes. The obtained attachment distance undergoes additional discretization before it is assigned as a feature, but we omit the details regarding the concrete discretization routine due to space limitations.

In the below description, the term "context features" is introduced as a convenient means of referencing the characteristics of a group of ordered word nodes: the node corresponding to a word at a given sentence position, and the nearest two word

nodes to its left and right (i.e., context windows always span over 3 adjacent word nodes).

The complete list of features for each edge type follows:

1. Word \rightarrow sentence root: attachment distance; node types; POS tag candidates (word nodes only); context word strings (word nodes only); conjunctions between: (i) the attachment distance feature and the node type and candidate POS tag features; (ii) the word node's string and its corresponding POS tag candidates; (iii) the POS-tag candidates in the word's context window.

2. Word \rightarrow word: attachment distance; node types; POS tag candidates; context word strings (head and dependent are modeled separately); conjunctions between: (i) the attachment distance feature and the node type and candidate POS tag features; (ii) the head and dependent nodes' word forms; (iii) the candidate POS tags of the head and dependent nodes and their context; (iv) the context words of the head and dependent nodes; (v) the word strings and the POS tag candidates of the head and dependent nodes.

3. Co-reference \rightarrow word: node types; word string; POS-tag candidates for the corresponding word form.

4. Co-reference \rightarrow co-reference: attachment distance; node types; POS tag candidates; context word strings (head and dependent are modeled separately); conjunctions between: (i) the attachment distance feature and the node type and candidate POS tag features; (ii) the head and dependent nodes' word forms; (iii) the candidate POS tags of the head and dependent nodes and their context; (iv) the context words of the head and dependent nodes; (v) the word strings and the POS tag candidates of the head and dependent nodes.

5. POS candidate \rightarrow word: node types.

6. Correct POS node \rightarrow candidate POS node: node types; context word strings; context POS-tag candidates; conjunctions between: (i) the context words; (ii) the POS tag candidates in the word's context window; (iii) the word and its corresponding POS tag candidates.

5 Results and Discussion

Our dataset comprises 190,000 tokens from the dependency version of the BulTreeBank. Of these, we used 90% for training, and 10% - for testing. We compiled the two subsets by allocating every tenth sentence to the test split, and putting all re-

maining sentences into the training split.

We trained and evaluated two versions of the MSTParser using the original version of the algorithm (and tree representation model) that constitute our baseline results. For the first experiment, we excluded all available information other than the word forms, to observe an accuracy of 65.21% (LAS). Next, we incorporated the gold standard morphosyntactic tagset of BulTreeBank, and noticed a dramatic increase in accuracy – 83.93% (LAS) for dependency parsing.

Georgiev et al. (2012) reported POS tagging accuracy between 95.72% (for guided learning without added linguistic resources) and 97.98% (for guided learning with an inflectional lexicon and applying linguistic rules over the output). In order to provide a meaningful comparison to the results yielded by our system on the dependency parsing subtask, we trained a separate model in a pipeline-like setting, using the predictions of the best tagger model described in (Georgiev et al., 2012).

When given the gold standard POS tags as input, the described dependency parsing algorithm yielded 87.6% LAS. However, training it with predicted POS tags decreased its accuracy to only 82.1% LAS against the test set for the joint task, owing to the errors of the tagger component.

We evaluated the proposed joint model through a number of experiments, whose results are summarized in Table 1. Its first instantiation took into account the word forms and the tags predicted by the inflectional lexicon, and excluded all features modeling the word context and all feature conjunctions. It yielded 95.99% (Accuracy), 46.96 (F) and 81.22% (LAS) for the POS tagging, co-reference and dependency parsing respectively (line 1 in Table 1).

As the sparseness of observations stemming from the great number of POS tags available in the BulTreeBank may lead to various issues, we attempted a different approach for handling the POS tags. We decomposed them to atomic characteristics – such as the part of speech and grammatical features such as person, gender, and number – that convey the meaning of the complete tags. We replaced the POS tag features incorporated in the first model with the new set of features that reflects their atomic counterparts, and repeated the experiment. However, we observed a small drop in accuracy (line 2).

We continued experimenting by complementing

our first model with word context features (line 3). For our next model, we revised the graph distance features, and stopped accounting for service nodes in their computation (line 4). Following that, we added all conjunct features, including combinations between the head and dependent morphosyntactic tags and the bigrams generated over the context of the head and child nodes' words (line 5, respectively). Line 6 shows the results yielded after adding the full set of conjunctions between the POS candidates and the wordform strings of the head and child nodes. Using this final feature set, we obtained the highest scores of 97.13% and 82.39% for POS tagging and dependency parsing respectively. However, the F-score computed for co-reference results decreased for this feature set.

At the dependency parsing task, we achieved a dramatic improvement over the scores yielded by our baselines, and slightly outperformed the pipeline-based model described earlier. Our results for the POS tagging task are aligned with the current state-of-the-art for Bulgarian. However, a direct comparison to (Georgiev et al., 2012) is not possible, since their POS tagging component was trained on the morphosyntactic subset of BulTreeBank that is two times larger than the dependency subset we used, and it was evaluated against a different collection of test sentences.

Our results for the co-reference subtask are in line with the results reported in (Recasens et al., 2010) for other languages. Our dataset is bigger than the datasets for Dutch, English and Italian, and similar in size to the datasets for Catalan and Spanish. The annotations available in our dataset are also comparable to theirs: POS, morphosyntactic information, heads, dependency relations, named entities, etc. However, semantic roles are missing in BulTreeBank. Our experimental settings resemble the singleton co-reference settings described in the cited work. If we take F-measure as a comparison criterion, our results (51%) are similar to the results for Catalan (56.2% SUKRE²), Spanish (55% SUKRE), Italian (50.4% SUKRE). We mention these results only for illustrative purposes, and this comparison has no pretension for completeness. However, in our case the precision is very high (around 80%), while the recall is low (around 30%). It should be noted that in (Recasens et al., 2010) balanced values prevail, and recall usually dominates precision. Our con-

²SUKRE is the system that performed the task.

clusion is that the features included in our model need to be carefully revised.

6 Conclusion and Future Work

The results reported in this paper indicate that three core tasks, namely POS (morphosyntactic) tagging, co-reference resolution and dependency parsing, can be solved via a combined model based on the MSTParser. Our approach is language independent. The model depends on the availability of a dependency treebank with annotated co-reference chains and morphosyntactic information. The model would be better manageable if the number of the possible POS tags for each wordform remained small. In our experiments we use a morphosyntactic lexicon and a guesser. Thus, we expect similar resources to be available for other languages. We expect also some of the interactions observed for Bulgarian to hold for a number of other languages, at least with respect to the connection between phenomena like binding, control, pro-drop, on the one hand, and rich morphology, on the other. Since the co-reference might be dependant mainly on morphological features (in morphologically rich languages) and/or syntactic positions and dependencies (both - in morphologically rich and morphologically poor languages), the difference would be rather explicated in the degrees of mutual interaction. Our expectation would be that the morphologically poorer the language, the bigger role of the word order and syntactic dependencies.

The joint model achieves performance similar to that of the current state-of-the-art for the POS-tagging task, and the combined model outperforms the dependency parsing in the pipeline currently available for Bulgarian.

The features used for single-task modeling cannot be easily ported to the joint modeling setting, and further design and experimentation with regard to the feature sets are required in order to improve the performance of the system. Such an effort may as well support the incorporation of other tasks in the proposed joint modeling framework. Some ideas we have in this regard include the addition of semantic class annotations to the individual wordforms, as well as features derived by some form of shallow analysis, such as chunking. We expect that such extensions will improve the performance of the system with respect to the dependency and co-reference resolution tasks.

Still, in future work we plan to attempt modeling the three tasks via a transition-based model that will require the simultaneous consideration of more than two non-adjacent nodes in the sentence. For example, in the Bulgarian sentence: “Toj μ_1 ya₂ podade kartinata₂ na Ivan₁.” (“He him it gave picture-the to Ivan”, *He gave the picture to Ivan*), a co-reference chain exists between the dative clitic ‘mu’ and the person name ‘Ivan’ which is interacting with the dependency relations between the clitic, the proper name, and the verb ‘podade’.

We also intend to experiment with alternative encodings of the co-reference chains in order to achieve a better use of the information available in our resources. Another direction of future work is the application of the described approach to treebanks of other languages.

Acknowledgments

The contribution of Kiril Simov and Petya Osenova is partially supported by the FP7 Capacity Project AComIn: Advanced Computing for Innovation (316087).

References

- Bernd Bohnet and Joakim Nivre. 2012. A transition-based system for joint part-of-speech tagging and labeled non-projective dependency parsing. In *EMNLP-CoNLL*, pages 1455–1465.
- Jenny Rose Finkel and Christopher D. Manning. 2010. Hierarchical joint learning: Improving joint parsing and named entity recognition with non-jointly labeled data. In *Proceedings of ACL 2010*.
- 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.
- Ryan McDonald, Koby Crammer, and Fernando Pereira. 2005. Online large-margin training of dependency parsers. In *Proceedings of ACL’05*, pages 91–98.
- Xian Qian and Yang Liu. 2012. Joint chinese word segmentation, pos tagging and parsing. In *EMNLP-CoNLL*, pages 501–511.
- Marta Recasens, Liu Màrquez, Emili Sapena, M. Antònia Martí, Mariona Taulé, Véronique Hoste, Massimo Poesio, and Yannick Versley. 2010. SemEval-2010 task 1: Coreference resolution in multiple languages. *Journal of the Association for Computing Machinery*, 28(3):114–133.