



## Sculpting Enhanced Dependencies for Belarusian

---

Yana Shishkina and Olga Lyashevskaya

EasyChair preprints are intended for rapid dissemination of research results and are integrated with the rest of EasyChair.

July 10, 2022

# Sculpting enhanced dependencies for Belarusian

Yana Shishkina<sup>1</sup> and Olga Lyashevskaya<sup>1,2</sup>

<sup>1</sup> The National Research University Higher School of Economics, Myasnikitskaya ulitsa  
20, Moscow, 101000, Russia

`yaashishkina@edu.hse.ru`, `olesar@yandex.ru`

<sup>2</sup> Vinogradov Russian Language Institute RAS, Moscow, Russia Volkhonka Street  
18/2, Moscow, 119019, Russia

`olesar@yandex.ru`

**Abstract.** Enhanced Universal Dependencies (EUD) are enhanced graphs expressed on top of basic dependency trees. EUD support representation of deeper syntactic relations in constructions such as coordination, gapping, relative clauses, argument sharing through control and raising. The paper presents experiments on the EUD parsing of the low-resource Belarusian language, for which no corpus with enhanced annotations was available.

Models trained on the Universal Dependencies treebanks of two closely related Slavic languages, Russian and Ukrainian, were used to parse sentences translated from Belarusian. After that, enhanced dependencies were projected to the original sentences, which gave us ELAS (Enhanced Labeled Attachment Score) 78.1% for both Russian and Ukrainian in evaluation. We also trained the model of one of the IWPT 2020 Shared Task participants on obtained annotations in Belarusian and achieved ELAS 83.4%. Analysis shows that the most common mistakes of cross-lingual parsing root in different theoretical perspectives and practice approaches to the annotation of particular types of clauses in the three Slavic treebanks. Both Russian and Ukrainian EUD transfer models tend to make mistakes when dealing with the predicate argument relations, which are hard to identify without understanding the semantics of the sentence. The alignment method decreases the quality of the annotation by confusing tokens that occur in a sentence more than once.

**Keywords:** dependency parsing · enhanced dependencies · Universal dependencies · annotation projection · Belarusian.

## 1 Introduction

Enhanced dependencies were introduced to Universal Dependencies (UD) in [1], as deeper syntactic relations which can't be represented by a syntactic tree structure and require a graph. Four main goals of the enhanced universal dependencies (EUD) include adding ellided nodes, propagating relations to conjoined tokens, adding the subject to controlled verbs and specify the information about case, prepositions or conjunctions where needed.

Not all of the UD treebanks have all or even some types of enhanced dependencies annotated. Two East Slavic treebanks, Russian and Ukrainian, have such annotations; they were improved recently during the IWPT 2020 [2] and 2021 [3] Shared Task. Our goal was to create enhanced annotations for Belarusian, a language closely related to Russian and Ukrainian. In this paper we compare several approaches, including making use of data of a related language, and creating rules that take into account basic Belarusian syntactic dependencies and training a model on top of obtained Belarusian annotations.

## 2 Related work

Two methods of enhanced dependency annotation, rule-based and data-driven, were compared in [4]. The research showed that both approaches are applicable for annotation, but the scores of the second approach may be increasing with the introduction of multilingual transformer models for retrieving word embeddings. The majority of IWPT participants such as [5–7] and others, chose to use data-driven methods. Hybrid methods that use rules are still able to show high quality as was proven in [8]. However, all kinds of methods encounter some difficulties while parsing constructions such as coordination, control & raising, and relative clauses [3].

A methods of cross-lingual dependency parsing within closely related language groups were surveyed in [10], for West and South Slavic languages, and in [9] for Scandinavian languages. In [9], authors suggested two different ways of using related language data, delexicalization and annotation projection, with the second approach showing better results. Our work uses an annotation projection very similar to one described in [9], but it is aimed at enhanced graphs instead of base dependency trees.

## 3 Method

### 3.1 Rules

We prepared the gold standard by annotating the Belarusian treebank with relatively simple rules, which were similar to those described in [8].

- Tokens with `advcl` and `acl` base dependencies, which denote clausal modifiers of nouns and predicates respectively, get additional information about conjunction, which is expressed by `mark` dependent. If conjunction has `fixed` dependent, i.e. is a complex (multi-word) conjunction, it is taken as a whole.
- Prepositions are added to oblique nominals and nominal modifiers (`obl` and `nmod`) by the same scheme, but instead of `mark` lemma is taken from `case` dependent.
- `obl` and `nmod` tokens also get additional information about their case, which is usually extracted from token grammar. If there is `nummod:gov` dependent, i.e. the token has a quantifier, case is extracted from quantifier.

- All conjoined tokens marked with `conj` dependency also get the same dependency as the first token in the sequence.
- If a verb has a controlled verb (`xcomp` dependent) and a subject, then the relation between the subject and the controlled verb is also added.
- `Ref` dependency is added to the tokens in a relative clause that have `Pron-Type=Rel` grammatical feature or that belong to the specific set of Belarusian words and the specific part of speech.

We decided to leave the annotation of ellipsis for future research because it is an extensive and self-sufficient work to design rules for the elided node retrieval in a new corpus.

### 3.2 Cross-lingual transfer

**Translation** We compared three machine translation services that support Belarusian, Google translate [11], Yandex translate, and Apertium [12]. We manually checked three variants of translations of 50 sentences with length above average and found that Yandex translate suits our task best. Apertium does not translate out-of-vocabulary words, and our task does not allow leaving some tokens untranslated. Google translate tends to drastically change the structure and word order of the sentence, which can decrease the quality of the alignment.

**Annotation of the translated sentences** The sentences were annotated with basic dependencies using UDPipe [13]. To add enhanced annotations, we chose Alibaba-NLP [14], a model which showed ELAS (Enhanced Labeled Attachment Score) 92.3% for Russian and ELAS 88.0% for Ukrainian in the IWPT Shared Task 2020 coarse post-evaluation.

**Alignment** There are some tools for language alignment such as *simalign* [15]. However, experiments on a sample of sentences showed that these tools have a high probability of not giving a word any pair, which is inconvenient, since we know that the three chosen languages are closely related and there are rarely words in one language that will be represented by none in the other.

We aligned tokens based on the cosine similarity of their mBERT [16] embeddings. Next we had to align enhanced dependencies to the original sentences. If there were more than one source language token aligned to the target, we chose the dependency of the head in the established group. If a syntactic group was not established we chose the token with maximum cosine similarity. Conversely when there were less tokens to align, we could assign dependency to the head of the Belarusian group but had to copy the base dependency to other tokens.

**Additional rules** It is clear that conjuncts and prepositions are different in all three languages, so the transfer doesn't help with enhancement of oblique nominals *obl*, nominal modifiers *nmod* and clausal modifiers *advcl*, *acl*, which require lemma of dependent conjunct or preposition added to the dependency.

To add this enhancement we used the same rules as we used to create the gold corpus. We applied a set of rules to fix the obvious errors:

- punctuation is always labeled *punct*;
- a sentence must have a *root*, i.e. the head of the sentence;
- a token cannot have itself as a head.

For Ukrainian as a source language, language specific relation subtypes such as *xcomp:sp* denoting secondary predication or *nsubj:rel* which is used for the subjects of relative clauses were converted to basic labels.

### 3.3 Training Belarusian model

We used Alibaba-NLP system, one of IWPT 2020 participants, to train Belarusian model on data annotated using above rules. The Alibaba-NLP team used neural networks for predicting if the relation is present between a pair of tokens, which were represented by multilingual contextual embeddings XLMR. Neural networks were also used to predict the type of the relation. As the result a connected graph with the most probable relations is chosen.

IWPT 2020 participants aimed to create a universal tool for the majority of languages, so no changes were needed to train the model for annotating enhanced dependencies in the Belarusian treebank. Annotated with rules Belarusian data was used for training the model.

## 4 Data

We used the following three UD treebanks.

**UD-Russian-SynTagRus** v.2.7 (1106k tokens, 62k sentences), automatically converted to UD [17];

**UD\_Ukrainian-IU** v.2.7 (122k tokens, 7k sentences) with native UD annotation [18];

**UD\_Belarusian-HSE** v.2.8 (305k tokens, 25k sentences) with native UD annotation.

Enhanced dependencies in the development part (1300 sentences) of the Belarusian treebank were checked manually and used for the analysis.

## 5 Results

Two metrics are used to evaluate enhanced dependencies. The ELAS (Enhanced Labeled Attachment Score) is F1-metric over arcs and all labels. The EULAS is F1-metric over arcs and universal for all languages labels, so EULAS does not consider errors in language specific labels.

The ELAS and EULAS scores are shown in the Table 1. The results of the transfer from both languages are very similar. The model trained on the rule-based annotated Belarusian data shows better results than both transfer methods. In Section 6, we will consider the sources of errors that cause the transfer approach to have lower quality than the the within-language model.

**Table 1.** Results

<b>Method</b>	<b>ELAS</b>	<b>EULAS</b>
Russian-Belarusian transfer	78.13	79.80
Ukrainian-Belarusian transfer	78.11	79.74
Belarusian: rules + AlibabaNLP	83.43	84.86

We also estimate the precision of enhanced annotations *sensu stricto* not taking into account those that are copied from basic dependencies. The ratio of dependencies adding new information to the base structure is approximately 30% in all of the three outputs. We calculated the number of EUDs obtained by transfer that matched EUDs in the gold annotations of the treebank and divided it by the number of all enhanced dependencies that do not copy basic ones. The results for Russian-Belarusian and Ukrainian-Belarusian transfer are nearly the same: 0.544 and 0.541 respectively. The metric for the Belarusian model is 0.6. These results confirm that the model trained on Belarusian data performs better than transfer and suggest that it is more difficult to choose the correct label of EUD than identify whether additional dependency is needed at all.

## 6 Analysis

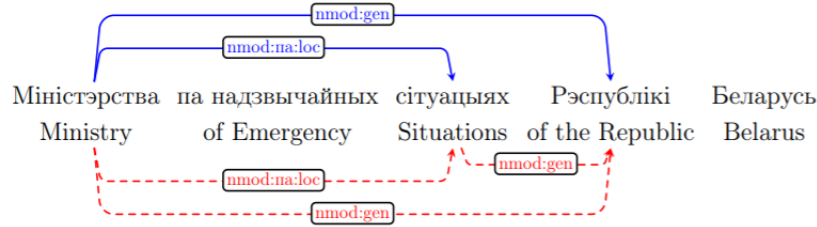
We divide the mistakes into three groups by their cause. First, we will analyze the errors which were caused by the unique features of a specific treebank. Then we will describe the mistakes which were mostly the expenses of the model or the transfer method.

### 6.1 Common mistakes for Russian-Belarusian and Ukrainian-Belarusian transfer

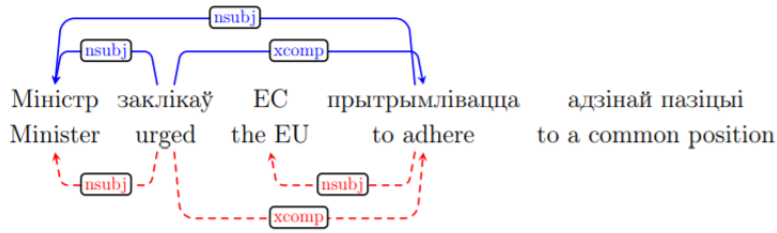
Rules that we used for annotating Belarusian data add a relation between all of the conjoined tokens and the head of the first token in sequence. Analyzing the differences between rule annotated and transfer annotated data we found out that with transfer annotation conjoined tokens also get the relations with the children of the first token in sequence, not only its head. It seems to us that this additional rule is useful for enhanced annotation and avoiding it in our rules was a mistake.

Transfer approach also revealed that in the Russian and Ukrainian treebanks, modifiers such as `nmod` get the same dependency as its head, i.e. establish the relation between its heads head, as shown on Fig. 1. The benefits of this rule are not completely clear to us, so adding it to our set of rules is open for discussion.

Our rules help to create a relation between a controlled verb (`xcomp`) and its semantic subject, which is the same as the syntactic subject of its head verb. It is quite hard to detect whether token is a semantic subject or an object of the controlled verb, so we chose to always use the most probable alternative. Clearly it causes some amount of mistakes, but using the model to add the relations

**Fig. 1.** The relation between `nmod` and its head head

between controlled verb and both its objects and subjects does not help to avoid this problem completely. Both Ukrainian and Russian model tend to confuse object and subject roles when applying this rule. An example is shown on the Fig. 2.

**Fig. 2.** The relation between controlled verb `xcomp` and its semantic subject

There are also some less important mistakes, since both of the alternatives can be chosen without violating the enhanced UD rules.

The inventory of multi-word prepositions and other bound expressions differs among the three treebanks. As a result, the relation `fixed`, which is used to link words in such syntactic idioms, is not always present in the representation of one language whereas it is there in another in the translation of the same phrase.

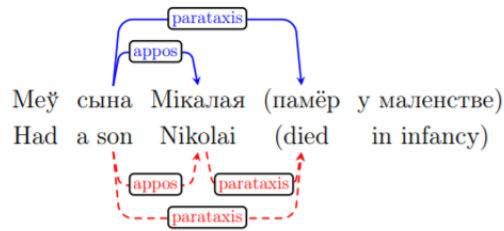
The heads of clauses which come after a dash or semicolon, and clauses in parentheses are usually marked as `parataxis` in the Belarusian treebank, but in Russian and Ukrainian they are usually appositional modifiers `appos`. These dependency types show very similar relations according to the UD guidelines, so the confusion of these types will not make a great difference to the understanding of the syntactic structure.

URLs and some emphatic text decoration HTML-tags are deliberately preserved in the texts of the Belarusian treebank, being attached with a special tag for unspecified dependencies, `dep`. After applying the UED transfer the tags get the relation `punct`. Even though HTML-tags often have similar functions as punctuation, it is better to separate those two kinds of tokens.

## 6.2 The mistakes of Russian-Belarusian transfer

Not only modifiers and conjoined tokens get the relations of their head in the Russian treebank. Tokens with **appos** are also connected to its heads head when the enhanced annotation is applied as on the Fig. 3.

Fig. 3. The relation between **nmod** and its head head



Dependency **acl** is never used to mark the head of the relative clause in the Russian treebank, but it can be seen in some cases in the Ukrainian and Belarusian treebanks. This fact can influence the ability of the transfer method from Russian annotation to add **ref** dependency in all the expected cases in Belarusian. In our opinion, it would be better to unify all of the relative clauses under one dependency **acl:relcl** as it is done in the Russian treebank.

Numerals in parentheses have a **nummod** relation with its head, while in the Belarusian treebank they have the same relation as other clauses in parentheses - **parataxis**. Russian variant is not always accurate since the date or the year in parentheses is not a numeric modifier of its head.

## 6.3 The mistakes of Ukrainian-Belarusian transfer

There are participles that have a **amod** dependencies in Ukrainian but their translations in Belarusian would have an **acl** dependencies. This is related to the fact that in the Ukrainian treebank participles are considered as adjectives and whereas they are verbs in Belarusian and Russian data. So it is clear that adjectives would be adjectival modifiers and verbs would be heads of the clauses.

In the context of passive voice an auxiliary verb in the Belarusian treebank is marked as **aux:pass** since its function is to demonstrate some grammatical features of the main verb. After Ukrainian-Belarusian transfer it is common to find a **cop** dependency in the same context. This kind of dependency is used as a linking verb with non-verb predicates. The cause of this difference is similar to the previous one and has to do with the part of speech of the participles in different treebanks.

In Ukrainian the word *ščo* 'that' is not considered a referent word in relative clauses, so it does not get **ref** dependency, it is always seen as **mark**. This does not apply to Russian or Belarusian.



One more small but interesting difference is that in the Ukrainian treebank, conjunctions can be the parts of conjoined sequence, too.

#### 6.4 The mistakes of alignment method

Some mistakes occurred because we chose to use the cosine similarity and some of the words vector representations in two languages were similar despite them not being the translations of each other. Most common of them is that punctuation has the wrong head after the transfer.

The same punctuation sign can be used more than once in one sentence and their vectors would be almost the same, so it is clear why their dependencies are confused while aligning the sentence and its translation. Similar mistake happens with other tokens, which tend to be used frequently in one sentence, such as tokens of relations `case`, `mark`, which denote prepositions and conjunctions.

Sometimes it does not have to be exactly the same word form to confuse the alignment script. Dependencies can be assigned wrong for the different forms of the same lemma. We could have avoided this if we used grammatical information that already existed, because we had no goal to convert raw text to enhanced dependency trees.

Our analysis shows that alignment is a part of transfer that is most responsible for producing mistakes. The method of alignment consists of several systems and each of them has its flaws.

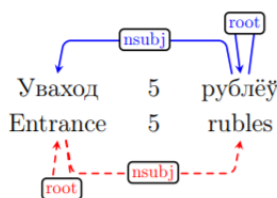
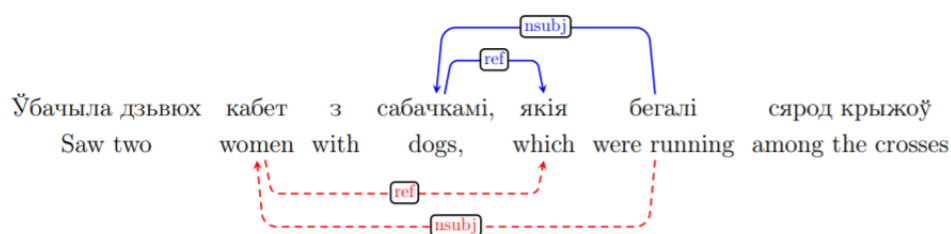
Although it is difficult to report an exact ratio of the alignment mistakes due to the fact that the same kind of mistake can be caused by multiple factors, we can roughly estimate their contribution. In general, the wrong choice of the dependency head accounts for ca. 60% of all mistakes. About one half of such mistakes can be explained by the alignment issues. A group of errors in which the predicted label lacks one or more dependency consists almost entirely of alignment mistakes. Summarizing we can say that nearly 40% of all errors are caused by inappropriate alignment.

#### 6.5 The mistakes of the model

Part of the differences was not caused by language features or alignment errors, they occurred because the annotation model is not able to predict everything perfectly.

The models for both Russian and Ukrainian tend to choose subject wrong in nominal clauses. It is not always clear what is a predicate and what is a subject in such clauses even for a person, so this kind of contexts is a problematic place for the automatic annotator. One of such cases is illustrated on the Fig. 4.

The models often fail to choose the right subject or object for controlled verbs `xcomp`, because the context can be ambiguous and that can not be fixed without understanding of the semantics. Same also applies for the contexts when there must be found a referent for a relative clause - there is not always only one candidate for that role, as on the Fig. 5.

**Fig. 4.** The confusion between `root` and `nsubj`**Fig. 5.** The incorrectly chosen `ref`

## 6.6 The mistakes of the model trained on Belarusian data

We analyzed the differences between the rule annotated data and the data that was annotated by the model trained directly on Belarusian data.

The most common difference was in the head of the punctuation. There are some alternatives: punctuation can depend on the head of the clause or its last token; it can be dependent of the previous clause or the next one, etc. The model chose different alternatives from those in the original Belarusian treebank, but we do not consider this a problem for the understanding of the syntax of the sentence. It is also quite interesting, that punctuation can have more than one head in the model annotation, although it does not occur in any of the Slavic treebanks.

As we mentioned earlier, our rules do not add the relation between all of conjoined tokens and dependents of the first one, but the model does. Moreover, the relations between modifiers such as `nmod` and `amod` and children of their head are added by the model.

We discovered that model struggles to detect bound expressions and does not add the relation `fixed` in all the places needed.

The rules are more successful in adding cases, prepositions and conjunctions, since it is impossible to extract wrong information if it is given. The model predictions are not so accurate and the information can be missed. However, the model is able to predict the case of the words, that do not have this grammatical information, such as abbreviations.

## 7 Discussion and concluding remarks

We have run three experiments: transferred enhanced dependencies from Russian and Ukrainian into Belarusian and annotated enhanced dependencies by the model trained directly on the Belarusian data and compared their results.

The transfer method underperforms the trained model by 5%. The performance of the transfer heavily depends on the accuracy of alignment. Other methods of alignment can be applied in the future.

The limitation of the transfer method lies in the similarity of the language structures and annotations in general. Transfer is best applied to languages with similar syntactic structure, such as three Slavic languages in our case. Although these languages are closely related, annotation in their UD treebanks differ in detail. Inconsistent use of tags such as the relative clause relation `acl:relcl` in the Ukrainian and Belarusian treebanks can be avoided in native basic UD annotation. There are some differences in views on the annotation of certain linguistic phenomena in Slavic languages, which should be considered by the corpus developers and researchers in the future. Such cases include coverage of multi-word expressions, the choice of a part of speech for participles, and more.

Nevertheless, with small adjustments, cross-lingual parsing can be seen a reasonable way of creating the native EUD annotation for Belarusian, for which deeply annotated corpora have not been available so far. This method can also be leveraged in the annotation of the basic UD structures.

Training the model on Belarusian data and using it for annotation showed to be a more efficient method than transfer. The structure and origins of mistakes are not always clear. The annotation with the transfer can be adjusted more easily.

The specific of enhanced dependencies is their diversity, so defining the set of enhancement rules was not an easy task and it is still open for discussion. In our work we studied some of the ambiguous cases of annotation of the Russian, Ukrainian and Belarusian treebanks. Most of the differences were not exactly mistakes but varied views on the language features. We also described some weaknesses of the alignment method and the annotator model. Taking these mistakes and differences into account can help to improve future works on the cross-lingual syntactic annotation.

The annotation resulting from our experiments was made publicly available in the UD\_Belarusian-HSE treebank v.2.8.

**Acknowledgments** This research was supported in part through computational resources of HPC facilities at HSE University [19].

## References

1. Schuster S., Manning C. D.: Enhanced English universal dependencies: An improved representation for natural language understanding tasks. Proceedings of the Tenth International Conference on Language Resources and Evaluation (2016).

2. Bouma G., Seddah D., Zeman D.: Overview of the IWPT 2020 shared task on parsing into enhanced Universal Dependencies. Proceedings of the 16th International Conference on Parsing Technologies and the IWPT 2020 Shared Task on Parsing into Enhanced Universal Dependencies (2020).
3. Bouma G., Seddah D., Zeman D.: From raw text to enhanced universal dependencies: The parsing shared task at IWPT 2021. Proceedings of the 17th International Conference on Parsing Technologies (2021).
4. Nivre J., Marongiu P., Ginter F., Kanerva J., Montemagni S., Schuster S., Simi M.: Enhancing universal dependency treebanks: A case study. Proceedings of the Second Workshop on Universal Dependencies (2018)
5. He H., Choi J.D.: Adaptation of multilingual transformer encoder for robust enhanced Universal Dependency parsing Proceedings of the 16th International Conference on Parsing Technologies and the IWPT 2020 Shared Task on Parsing into Enhanced Universal Dependencies (2020).
6. Kanerva J., Ginter F., Pyysalo S.: Turku enhanced parser pipeline: From raw text to enhanced graphs in the IWPT 2020 Shared Task. Proceedings of the 16th International Conference on Parsing Technologies and the IWPT 2020 Shared Task on Parsing into Enhanced Universal Dependencies (2020).
7. Dehouck M., Anderson M., Gómez-Rodríguez C.: Efficient EUD parsing (2020).
8. Heinecke J.: Hybrid enhanced Universal Dependencies parsing. Proceedings of the 16th International Conference on Parsing Technologies and the IWPT 2020 Shared Task on Parsing into Enhanced Universal Dependencies (2020).
9. Tyers F., Sheyanova M., Martynova A., Stepachev P., Vinogorodskiy K.: Multi-source synthetic treebank creation for improved cross-lingual dependency parsing. Proceedings of the Second Workshop on Universal Dependencies (2018).
10. Agić, Ž., Tiedemann, J., Merkler, D., Krek, S., Dobrovoljc, K., Moze, S.: Cross-lingual dependency parsing of related languages with rich morphosyntactic tagsets. Association for Computational Linguistics. (2014).
11. Wu Y., Schuster M., Chen Z., V.Le Q., Norouzi M., Macherey W., Krikun M., Cao Y., Gao Q., Macherey K., Klingner J., Shah A., Johnson M., Liu X., Kaiser L., Gouws S., Kato Y., Kudo T., Kazawa H., Stevens K., Kurian G., Patil N., Wang W., Young C., Smith J., Riesa J., Rudnick A., Vinyals O., Corrado G., Hughes M., Dean J.: Google’s neural machine translation system: Bridging the gap between human and machine translation. (2016).
12. Forcada M. L., Tyers F.: Apertium: a free/open source platform for machine translation and basic language technology. Proceedings of the 19th Annual Conference of the European Association for Machine Translation: Projects/Products. (2016).
13. Straka M., Strakova J.: Tokenizing, pos tagging, lemmatizing and parsing UD 2.0 with UDPipe. Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies. (2017).
14. Wang X., Yong Jiang Y., Tu K.: Enhanced universal dependency parsing with second-order inference and mixture of training data. (2020).
15. Sabet M.J., Dufter P., Yvon F., Schütze H.: Simalign: High quality word alignments without parallel training data using static and contextualized embeddings. (2021).
16. Devlin J., Chang M.W., Lee K., Toutanova K.: Bert: Pre-training of deep bidirectional transformers for language understanding. (2018).
17. Droganova K., Lyashevskaya O., Zeman D.: Data conversion and consistency of monolingual corpora: Russian UD treebanks. Proceedings of the 17th International Workshop on Treebanks and Linguistic Theories. (2018).

18. Kotsyba N., Moskalevskiy B.: Using transitivity information for morphological and syntactic disambiguation of pronouns in Ukrainian. *Visnyk Natsionalnoho universytetu "Lvivska politehnika". Informatsiyni systemy tamerezhi.* (2019).
19. Kostenetskiy P.S., Chulkevich R.A., Kozyrev V.I. HPC Resources of the Higher School of Economics. *Journal of Physics: Conference Series.* (2021). Vol. 1740, No. 1. P. 012050. DOI: <https://doi.org/10.1088/1742-6596/1740/1/012050>