ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное

учреждение высшего образования

Национальный исследовательский университет

«Высшая школа экономики»

Факультет гуманитарных наук

Образовательная программа

«Фундаментальная и компьютерная лингвистика»

Егорова Евгения Сергеевна

ИССЛЕДОВАНИЕ ТОЧНОСТИ РАБОТЫ МЕТОДОВ СНЯТИЯ МОРФОЛОГИЧЕСКОЙ ОМОНИМИИ ПРИ УЧЕТЕ ЧАСТОТНОСТИ ОМОНИМИЧНЫХ СЛОВ

Morphology Disambiguation Using Tokens Frequencies

Выпускная квалификационная работа студента 4 курса бакалавриата группы 181

Академический руководитель образовательной программы	Научный руководитель
канд. филологических наук, доц.	канд. технических наук, доц.
Ю.А. Ландер	Э.С. Клышинский
_____	_____
« » _____ 2022 г.	

Москва 2022

Table of contents

Abstract

The main topic of this paper is full morphological tagging, a well-explored yet still relevant NLP task. In spite of excellent results already achieved, there is still some space for improvement. In our research, we aimed to develop a technique that could enhance the performance of existing morphological taggers and then test it on the advanced BERT-based model from the DeepPavlov library that was claimed to reach state-of-the-art results among open-source systems. In order to accomplish the improvement, we tried to make use of the linguistic information, concerning different homonymy types and their correlation with tokens frequencies. To our mind, by training several classifiers on separate groups of tokens with different ambiguation properties and then combining these classifiers into an ensemble to form a final prediction we could have outperformed the initial model, as each separate classifier would better learn to deal with certain homonymy type. However, our hypothesis has not been fully confirmed: even though we managed to enhance the performance by applying the described technique for most of the cases, the improvement appeared to be insignificant and rather inconsistent. Furthermore, we did not detect any evidence of the model better coping with certain ambiguity properties; instead, other factors were revealed that could be accounted for the quality increase.

1. Introduction

Starting from the inception of computational linguistics, grammatical ambiguity has always been an issue to consider, and even nowadays it continues to be a challenge for Natural Language Processing (NLP). This problem affects many NLP tasks, such as Morphological Tagging, Named-Entity Recognition, Machine Translation, etc.; thus, by increasing the quality of grammatical ambiguity one can improve the results of his or her main task.

In search of a solution to this problem, it is essential to understand that words can be ambiguous on several levels of language. One word may bear completely different meanings yet share the same part-of-speech and have the same forms in an inflectional paradigm for all of its meanings. For instance, consider *лук* as 'an onion' and *лук* as 'a bow' in Russian: this is a case of semantic ambiguity, and it concerns only word senses. On the other hand, a token can represent an overlap of inflectional forms of different lexemes, like the Russian *три* as an imperative of verb *тереть* and *три* as a preposition, or of a single lexeme, like *мать* as a form of the nominative and the accusative cases. In the latter case, we are dealing with grammatical ambiguity which is the main concern of this paper.

As was found in (Klyshinsky, 2021), morphologically ambiguous words share different properties depending on their overall frequency. This fact can be taken into consideration while solving the grammatical disambiguation task, which, to our knowledge, has not been done yet. Thus, it is of interest to examine whether utilizing the correlation between the homonymy types and word frequency distribution helps to improve the performance of existing approaches.

In this study, we develop a new method that would increase the quality of grammatical disambiguation for Russian. Usually grammatical disambiguation is included in the process of morphological parsing and, even though most present morphological taggers excel at this task (Dereza et al., 2016), it still merits attention, as morphological tagging is often included in text preprocessing, and in case of an error it would propagate further. Based on the assumption that the nature of ambiguity varies depending on word frequencies, our main hypothesis is that we could increase the model performance by splitting data into several parts based on the frequency of tokens and then fitting different models on these parts separately. As a result, we expect to produce several classifiers, each assigned to better deal with a certain type of ambiguous words, and by combining them all we anticipate significantly outperforming the initial model.

2. Literature Review

2.1. Approaches to the morphological tagging task

Morphological tagging is a process of marking up tokens with their grammatical features, such as part-of-speech, gender, tense, etc. As a separate task, automatic morphological parsing was firstly approached for American English in the 1970s in (Greene & Rubin, 1971): the study was conducted on the freshly launched Brown Corpus (Kucera & Francis, 1967) and only part-of-speech tags were assigned. The model was based exclusively on the dictionary information and hand-written rules, describing possible part-of-speech co-occurrences. It had an accuracy score of approximately 70% and presented the rule-based approach to morphological tagging. Then, in the middle of the 1980s, Hidden Markov models (HMM), still common modern statistical model for sequence-labeling, came into use to disambiguate grammatical categories. One of the pioneers of this approach (Garside, 1987) achieved a score of 96-97% on the Lancaster-Oslo-Bergen Corpus of British English, using a similar tagset as in the previous research. Thus, the stochastic approach to the morphological tagging was formed, which then was augmented by the implementation of other techniques, mainly making use of statistics, frequency and probability of n-grams (Kumawat et Jain, 2015), such as Bayesian Models, Conditional Random Fields, Maximum Likelihood Estimation, Support Vector Machines, etc.

Both approaches bear their benefits and limitations. As mentioned in (Brille, 1992), stochastic morphological parsing techniques are easier to implement as "the necessary statistics can be automatically acquired and the fact that very little handcrafted knowledge need be built into the system". Yet these models usually encounter some data-collection issues as they require a large manually tagged corpus; they are also unable to deal with unknown words (Awwalu et al., 2020, p. 714). On the other side, rule-based approaches are more interpretable and do not need a large amount of data, though construction of a hand-written rule-based system is rather time-consuming, probably requires linguistic background and cannot guarantee "that every linguistic rule is captured in the rule construction" (Awwalu et al., 2020, p. 713). This setup yields hybrid approaches to morphological tagging, combining the initial two techniques. All in all, a choice of the methodology of grammatical parsing highly depends on the obtained data and language.

First morphological taggers for Russian were mostly dictionary-based, for instance, (Segalovich, 2003) and (Sokirko, 2004). With the arrival of the accessible annotated corpora, multiple researchers began to implement variations of HMMs and other stochastic-based

approaches, adjusting existing morphological taggers for Russian. For example, in (Sharoff et al., 2008) three statistical models, TnT (Brants, 2000), TreeTagger (Schmid, 1994) and SVM Tagger (Giménez & Màrquez, 2004), were trained on the Russian National Corpus. All models achieved rather high performance, up to 95.28% of accuracy score with full tagset by TnT tagger. However, these results are lower than the results of the same model for the English language: (Thede & Harper, 1999) reports an accuracy score of 96.9%. Despite possible stylistic differences in texts, the gap is quite significant. The same tendency is shared by other models: 92.56% by the TreeTagger, reported in (Dereza et al., 2016) for Russian, and 96.36% in the original paper (Schmid, 1994); 92.24% in (Sharoff et al., 2008) and 97.56% in (Gimenez et al., 2003) for Russian and English languages respectively by SVMTool.

An application of statistical models becomes complicated because of the rich morphology of Russian: in comparison to the English language, there are more grammatical categories in Russian, and derivational and inflectional inventories are larger, which leads to an extension of the tagset. Therefore, an implementation of stochastic techniques requires a larger corpus to provide relevant statistical information for all possible tags. Another features of Russian, which may cause issues for morphological parsing, are "free word order and regular homonymy between different forms of the same word,..., which cannot be resolved by the immediate context of the word", as noted in (Sorokin et al., 2017, p. 4). Thus, statistical models, based on the closest co-occurrences, encounter some limitations as they cannot capture the coordination between two tokens, located within 2-3 words from each other. Hence for Russian morphological parsing more sophisticated systems are required, utilizing either complex linguistic features or the power of deep learning algorithms.

One of the tracks of the "Dialogue 2017", MorphoRuEval-2017, focused on automatic morphological analysis methods for Russian. According to the final report of organizers, the top-ranked algorithms, proposed by the participants, formed two groups: the first group used deep neural network approaches and the second one tried to gather some linguistic information and use it as a feature (Sorokin et al., 2017). The best accuracy score of 97.11% was achieved by the ABBYY team in the open track, who used a two-layer bidirectional neural network with several additional layers as a learning method (Anastasyev et al., 2017). However, two top-ranked models on the closed track utilized more linguistically-oriented approaches.

A subtask of the GramEval-2020 was also devoted to full morphological tagging (Lyashevskaya et al., 2020). Analyzing its results, one can notice that the top two teams used

BERT (Devlin et al., 2018) architecture in their algorithms and significantly outperformed other participants. One of the teams provided a comparative overview of their experiments with LSTM-, ELMO- and BERT-based models, which clearly demonstrates the advantage of BERT, surpassing other models in every experimental setup and reaching an accuracy score of approximately 98% on some subcorpuses.

Despite a high performance of the latter approaches, the organizers of GramEval 2020 still report the remaining difficulties in resolving pos-tagging homonymy as well as common errors with assigning morphological features. Plenty of errors concerns some high-frequency words, including *быть*, *что*, *значит*, etc. Thus, one could improve model performance by focusing on such cases.

2.2. Homonymy Types

In (Klyshinsky et al., 2015; Klyshinsky et al., 2020) classification of homonymy types was proposed. It includes the following categories:

- unambiguous tokens (norm): there is only one way of possible morphological analysis (like *Наташа*$_{\text{NOUN}}$);
- ambiguous by parameters (par): forms share the same pos-of-speech and lemma, grammatical features differ (like *Наташи*$_{\text{NOUN}}$ - genitive singular form or nominative plural form);
- ambiguous by part-of-speech (pos): forms refer to the same lemma, but different part-of-speeches (like *ученый*$_{\text{ADJF}}$ *кот* vs *ученый*$_{\text{NOUN}}$ *пришел*);
- ambiguous by lemma (lem): forms share part-of-speech, but their lemmas diverge (*Александра* could be genitive of *Александр*$_{\text{NOUN}}$ or nominative of *Александра*$_{\text{NOUN}}$);
- ambiguous by both part-of-speech and lemma (pos_lem), neither of those overlap (for example, *ясно*$_{\text{ADVB}}$ vs *ясно*$_{\text{ADJF}}$);
- out-of-vocabulary tokens (*natasha*, *2022*, *пенить*).

According to investigations described in (Klyshinsky et al., 2015; Klyshinsky et al., 2020), the distribution of homonymy types highly depends on the frequency of words. In (Klyshinsky et al., 2021) authors point out that among the top 100 most frequent tokens almost half of the ambiguous words are ambiguous by part-of-speech; the percentage of this type of homonymy drastically decreases for less frequent tokens, while a ratio of the ambiguity by grammatical parameters increases (Figure 1).

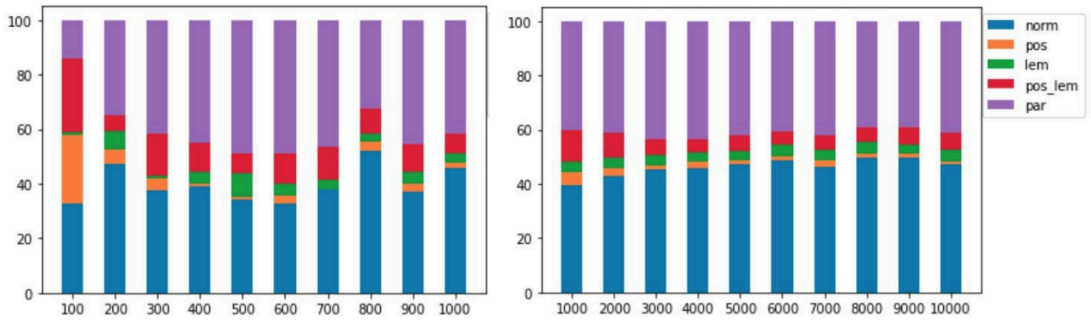

Figure 1. Distribution of Homonymy Types based on tokens frequencies for 1000 and 10000 most frequent tokens (from (Klyshinsky et al., 2021)).

One can see that the top 100 most frequent words have different grammatical ambiguity properties that differ from other frequent groups; the top 1000 most frequent tokens seem to differentiate from all the rest as well. However, all the tokens are analyzed simultaneously during morphological parsing. Thus, it is of interest to provide an approach, which would take into consideration the existence of different homonymy types, since, as noted above, plenty of errors of current state-of-the-art techniques come from the high-frequency words with differing ambiguation properties.

3. Methodology

One of the possible solutions to capture different ambiguity properties of words is to provide distinct classifiers for different homonymy types. However, it is impossible to define a type of ambiguity for the unlabeled data, therefore, it would not be clear which classifier should be applied. At this point, the observation that there is some correlation between word frequency and homonymy types comes in handy: in the case of partition by token's frequency each produced group would contain tokens that share ambiguity similar properties within the group but distinctive from the other groups; thus, we do not need texts to be tagged for such a split. These groups could be used for choosing a proper model focusing on a particular homonymy type, that is why there is no need for any additional tools or dictionaries excluding a raw text. We believe that by applying such a technique we would manage to integrate the knowledge of words ambiguation properties into an algorithm of morphological tagging and outperform the initial general model.

As a basic model architecture, we use morpho_ru_syntagrus_bert - BERT-based advanced model from the DeepPavlov library. This model has been chosen due to the fact

that it has open-source code and it is claimed to achieve state-of-the-art performance at the full morphological tagging task among open source systems (DeepPavlov, 2019).

The code was written in Python in Google Colab[1]. To train and evaluate models we utilized the TensorFlow and DeepPavlov frameworks. The performance of the models was evaluated with a per token accuracy metric.

Figure 2 illustrates a brief scheme of the proposed pipeline. Below it we provide some more detailed descriptions.

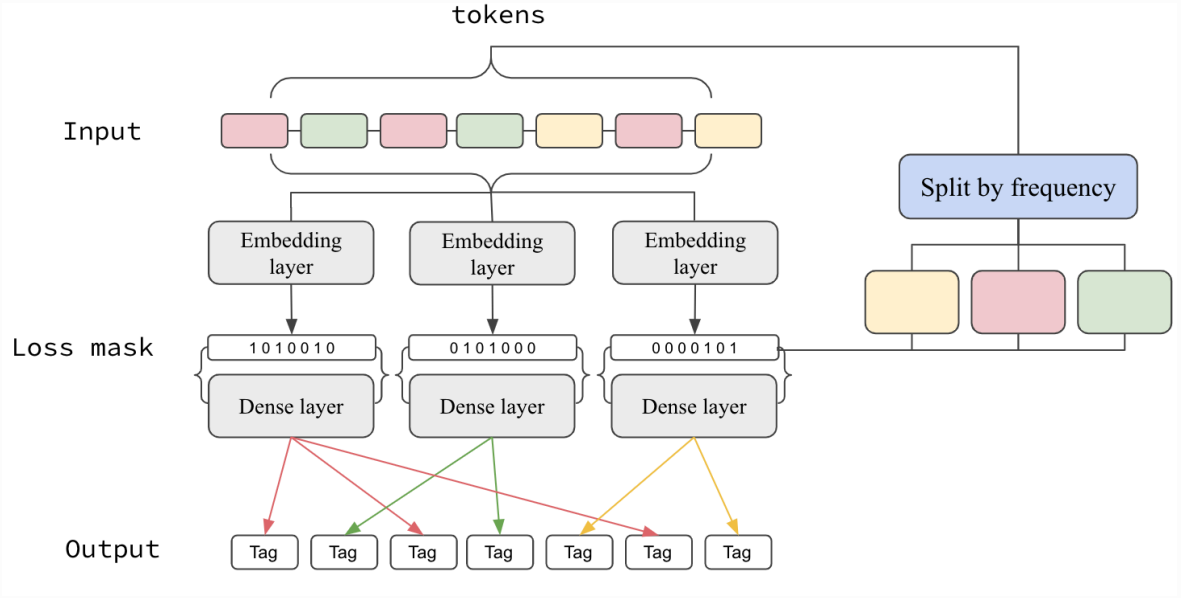

Figure 2. Pipeline summarization

3.1. Split tokens by frequency

At first, we split our input tokens into three groups according to their frequency: the first 100 tokens, tokens from 100 to 1000, and all the rest. The groups are formed based on the results of (Klyshinsky et al., 2021); in such a partition, homonymy types of tokens from each group seem to diverge at most for the corpora explored in the study. In this research, we use different datasets for our analysis but assume that the distribution of homonymy types would not significantly change; we also assume that the most frequent words tend to stay in the same group across different texts. At least, they were already classified by homonymy groups and we try to keep this division in our work. The frequency dictionary is built up based on just a train dataset; throughout the prediction, all out-of-dictionary tokens are included in the third group of the least frequent words. All the punctuation tokens are referred to the third group as well. It should be noted that during the calculation we stick to an

[1] The project code is available at https://github.com/jeka-e/morphotagging-with-tok-frequencies

approach proposed in (Klyshinsky et al., 2021): we consider tokens by their concrete instance, external form, not lemma, as was done during a similar analysis in (Klyshinsky et al., 2015).

3.2. Loss computation

After splitting the words by their frequency we fit three different models on each of the groups separately: every model receives information about the whole sequence but computes loss only by tokens belonging to a certain group. In such a way weights of each model are affected only by a certain group of words, which allows to capture more precisely its characteristics including, to our belief, their ambiguity properties. In order to achieve this effect, we create a special mask that has 1 at the positions of the tokens from a certain group and 0 at all the other positions. After this mask is fed to loss function as a weights parameter, each loss weight then is applied to the corresponding sample in batch. In the end, we implement an ensemble of three trained models, each being responsible for predicting tags for a certain group of tokens.

3.3. Tag dictionary

Morphological parsing consists of assigning a token's part-of-speech and its grammatical features, in different systems these predictions could be done either sequentially, one build upon the other (Qi et al., 2019), or simultaneously (Kanerva et al., 2018). DeepPavlov's BERT-based morphological tagger, the one we have chosen for our study, follows the second approach, it assigns part-of-speech tag and grammatical features at the same time. During the training phase, at first, a tag dictionary is formed: it includes all possible morphological tags for the words in the training dataset. Each tag represents a concatenation of part-of-speech and grammatical features of a token. To fit each of our classifiers, we use a general tag dictionary built up for the whole text. To our mind, producing separate tagset for different models would not affect its training. Each model takes into consideration a limited group of tokens in train data, so the tagset would only include the tags assigned to the tokens from this group. Tags that do not correspond to any of these tokens would not be included there and thus would not be predicted. But if these tags do not correspond to any tokens from the covered group, then, even in the case of including them into the tagset, for the associated classifier there would be no example of its occurrence in train data. Thus, the model would simply ignore them and not predict as well as in case of

providing a separate tagset - it seems to make no difference. So, for the sake of simplicity in some further experiments, we decided to use a general tag dictionary for every classifier.

3.4. Classification

To assign a tag to a token, the model performs multiclass classification with a single dense layer on the top of embedded tokens, as the embedder RuBERT-base-cased is used. Each tag is treated as a separate class. Under such a pipeline the number of classes is rather large, thus model "learns to ignore low frequent morphological features at the tail of the distribution" (Anastasyev, 2020, p. 2). Fitting models on different groups of tokens might lead to a reduction of classes for some classifiers as the tag diversity for certain tokens group is likely to be more limited in comparison to a general tagset. And the fewer there are classes, the more likely model would capture the rare ones. Thus, the performance could be affected not so much by the classifier adjusting to the particular homonymy types as by a simple decrease of classes. We could not come up with an idea of a pipeline to exclude this factor.

3.5. Experimental setup

The main goal of this study is to test whether the proposed modification enhances the performance of the initial classifier; our method does not imply altering the neural network architecture. Therefore, the general model comes out as a baseline. Our mission is to provide three fitted classifiers with the same architecture as in the baseline one to combine them in the ensemble and compare the quality. We set the hyperparameters to the default ones from a configuration file of morpho_ru_syntagrus_bert to keep them the same for a clear intercomparison of systems. The only mutable parameters were the number of epochs and batch size: we had to adjust them to the datasets due to their size difference. The patience limit was set to the value of 30: which means that after 30 iterations over batches with no quality improvement the training process would stop and the best-performed model would be saved. In this manner, we were able to set a rather big number of epochs to make sure the model learns as much as it could and meanwhile avoid overfitting. In hindsight, we should have captured both the last instance of weights and the best-performing ones, as the former might be better suited for the analysis of the model's behavior. But, as we are mainly focusing on the models' evaluation, such a decision does not really botch things up for our study but helps to prevent the problems of memorizing data and overfitting.

In order to test the proposed technique to the full extent, we conducted two experiments for each dataset. The first one implied training classifiers from scratch, for the

second we finetuned the general model. During each experiment, we produced three models - fitted on the first 100 most frequent tokens, on the 100 to 1000 tokens, and all the rest. Also, we trained a baseline model where necessary for further finetuning and comparison. Thus, we aimed to provide a total of 7 models or the majority of the datasets:

- 1 general baseline model;
- 3 trained from scratch models (model_top100, model_top1000, model_other);
- 3 finetuned models (model_top100, model_top1000, model_other).

However, in several cases, fewer models were produced. This happened due to the fact that during the finetuning experiments the quality often did not improve, thus baselines performed the best. And, as we captured only the highest quality instances of weights, there were no new models saved.

4. Data

4.1. Main Dataset

As the main dataset for training and testing we utilize the UD_Russian-SynTagRus subcorpus of Univeral Dependencies Corpora for Russian (version 2.3). This corpus has a consistent annotation of part-of-speech tags, morphological features and syntactic dependencies given in a CoNLL-U format. The training data includes over 1 M tokens, development and test sets, around 95 K tokens each. We chose this dataset as the main one for a few reasons: a) it has manually checked annotation; b) it is rather large and has open development and test sets of a considerable size which is essential for further error analysis; c) this dataset was used for the training and evaluation of BERT-based morphological tagger by DeepPavlov, which we use as the baseline model and a foundation for our classifiers.

4.2. Additional Datasets

We also conducted some experiments with some datasets from GramEval 2020 competition[2] in order to establish if applying the investigated method would achieve the same effect on the different corpora as on the main one. A choice of the GramEval 2020 data stems mainly from its availability, accurate annotation and popularity: apart from the track on Dialogue Evaluation 2020, it was also used as a benchmark for morphology taggers in Naeval project[3]. From the collection of GramEval we picked several datasets and formed two

[2] https://github.com/dialogue-evaluation/GramEval2020
[3] https://natasha.github.io/naeval/

different-sized subcorpora, each containing texts of a distinct genre - wiki and social media[4]. Hence, our sample is rather diverse, so we could test the flexibility of the proposed technique. As there was no test data shared, we used a development set for an evaluation.

A brief summary of the utilized datasets is provided in Table 1:

domain	train dataset	number of tokens	eval dataset	number of tokens
main	UD 2.3 ru_syntagrus-ud-train	1 M	UD 2.3 ru_syntagrus-ud-dev + ru_syntagrus-ud-test	95 K + 95K
wiki	GramEval2020-GSD-train	96 K	GramEval2020-GSD-wiki-train	1 K
social media	GramEval2020-Taiga-social-train + MorphoRuEval-VK-gold	31 K	GramEval2020-RuEval2017-social-dev	1 K

Table 1. The number of tokens and sources of the development and test sets in the additional collection.

5. Preliminary Analysis and Hypothesis

Before sharing the outcomes of our experiments we deem it necessary to formulate the hypothesis more clearly and provide some preliminary analysis of our sample for a more sensible interpretation of the results.

Predictably, the correlation between the homonymy types distribution and tokens frequencies retains for all of our datasets. So, our main expectation is an outperformance of the baseline: each classifier is expected to deal with a certain homonymy type in better quality, thus an ensemble should perform better. Taking a closer look, we expect the main improvement to come from the advanced POS-tagging for the most frequent tokens and an enhancement of the morphological features assignment for the less frequent ones. This assumption is based on the fact that these groups are rather distinct in terms of homonymy:

[4] In this paper we do not focus on exploring possible correlations between the performance of a morphological tagger and text genre. We did not include in our sample news and poetry datasets due to inability to collect enough annotated data, texts from the Middle Russian Corpus were considered too specific for our purposes, and we rejected GramEval2020-SynTagRus as we were already using another version of this dataset as the main one.

approximately half of the 100 most-frequent tokens that are homonymous share POS ambiguity, and another huge chunk of them are ambiguous by POS and lemma which also concerns differing parts-of-speech; as for less frequent tokens a ratio of the ambiguity by grammatical parameters drastically increases (Figure 3). Thus, if a model indeed uses different strategies to approach each homonymy type, then the classifier trained on the tokens with mostly POS-ambiguity should better tackle POS-tagging, and the same applies with regard to the grammatical features. In comparison to the tuning model on the whole text, the train data would not be so mixed up by homonymy types for a concrete classifier, so it might better catch the ambiguity properties of the tokens.

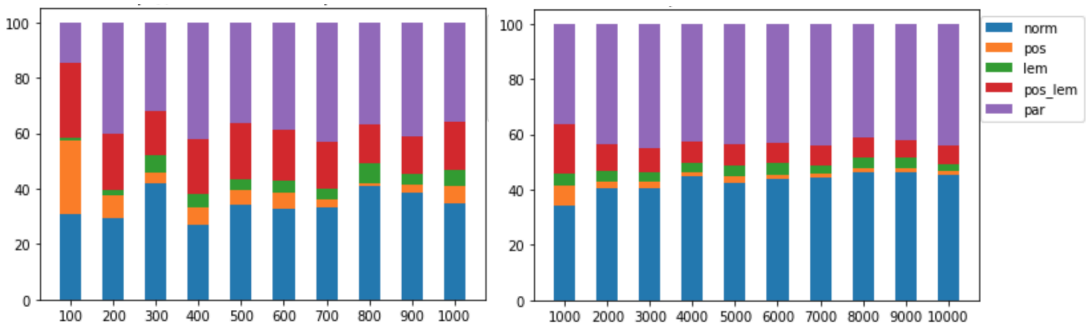

Figure 3. Distribution of Homonymy Types based on tokens frequencies for 1000 and 10000 most frequent tokens in the main dataset.

It is important to mention that the additional datasets have similar homonymy types distribution. We do not provide an illustration to confirm this fact but anticipate comparable results.

Furthermore, we expect the most improvement to come from the 1000+ by frequency tokens classifier. The reasoning is rather straightforward: the percentage of the tokens from this group is the highest - almost half of the total even without punctuation (Figure 4).

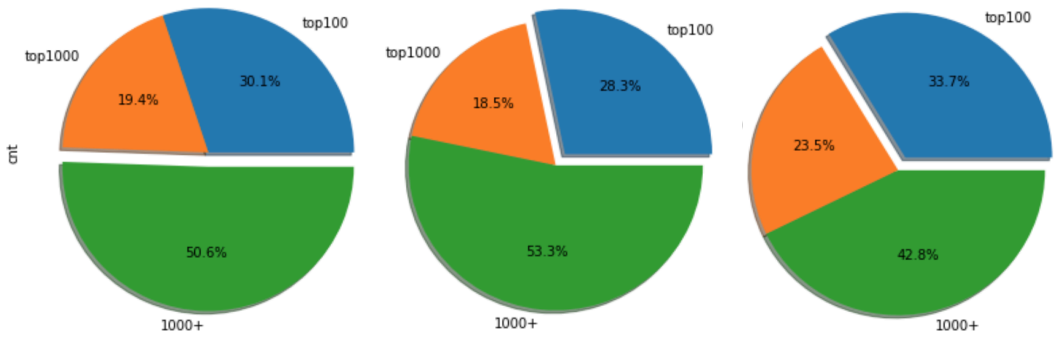

Figure 4. Token distribution by frequency group in SynTagRus, wiki, and social media datasets (without punctuation).

Assuming the less-frequent tokens share more of a grammatical parameters ambiguity, the overall performance should mostly correlate with the amelioration rate in morphological features assignment.

By splitting the training data in the suggested manner, we also partition the classes to predict for each classifier. Table 2 demonstrates the number of tags that correspond to tokens from every produced group.

dataset	top100	top1000	rest	general
main	137	287	634	733
wiki	163	304	554	666
social media	145	303	408	554

Table 2. The number of tags corresponding to tokens in each group.

As expected, across the groups this volume appears to be considerably lower, than in the general tag dictionary. At this stage, we could come up with two possible contradictory scenarios of it impacting the results. On the one side, by the decrease of classes and corresponding samples, we enhance the chances of the model focusing on rare morphological features and therefore improving the performance. But on the other side, in such a split we also remove from the training data the instances of the remaining classes as well, thus creating more low-frequent tags to ignore. Hence the consequences of class decrease are yet to be investigated.

6. Results and Discussion

6.1. Overall results

Table 3 represents the scores of the baseline model and our experimental systems on the main dataset. Tables 4 demonstrate the performance on the additional corpora. Apart from a general score, we detail the quality of the intermediary classifiers, the value is computed by the corresponding group of tokens. We underlined the result in the finetuning experiments if no new model was produced and the score just duplicates the baseline performance.

SynTagRus v2.3	experiment	model_top 100	model_top 1000	model_other	general
dev set	baseline	**0.9813**	**0.9783**	**0.9756**	**0.9774**
	from scratch	0.9812	0.9772	0.9745	0.9765
	finetuning	**0.9813**	**0.9783**	**0.9756**	**0.9774**
test set	baseline	**0.9852**	0.9796	**0.9752**	**0.9783**
	from scratch	0.9845	**0.9799**	0.9739	0.9774
	finetuning	**0.9852**	0.9796	**0.9752**	**0.9783**

Table 3. Scores of the systems on the main dataset

dataset	experiment	model_top 100	model_top 1000	model_other	general
wiki	baseline	0.9698	0.9477	0.9495	0.9538
	from scratch	0.9698	**0.9673**	**0.9572**	**0.9615**
	finetuning	**0.9741**	0.9608	0.9511	0.9577
social media	baseline	0.9112	0.8269	**0.919**	0.9027
	from scratch	0.9046	0.8397	**0.919**	0.9027
	finetuning	**0.9211**	**0.8590**	**0.919**	**0.9105**

Table 4. Scores of the systems on the additional datasets

As one could notice, there are not a lot of shared tendencies across the performance on the datasets. Rather, the results of the evaluation on the main sample and additional collection contradict each other: for the main dataset the best-performing system is almost always baseline, finetuning does not enhance the performance; the highest quality on the additional data, on the contrary, is achieved by the produced ensembles of classifiers. This phenomenon could possibly stem from the size difference between the samples. The poor performance of the baseline on the small datasets might result from the lack of enough data to

train a general classifier: as a tagset is rather diverse it focuses on the most frequent classes and sometimes ignores less common ones which occur just several times per training sample. On the contrary, while fitting the classifier on the limited group of tokens the chances of the model capturing these rare instances increase since the diversity of samples and corresponding tags in training data reduces. During the error analysis, we managed to reveal a few examples that go along with this theory. When examining the mismatches in the prediction made by the baseline model but not the ensembles, one could notice that several errors come from the cases of a model predicting the most frequent class while the second one is significantly less common; on the other hand, the separate classifiers do not make these mistakes. Nonetheless, we can not consider this remark to be more than just an observation as the number of these instances is pretty small and many of them include the annotators' mistake; the full proof would require a more many-sided analysis.

Between the experiments of finetuning the baseline and training classifiers from scratch, we could not determine the best approach. Both of them did not provide an improvement on the main dataset, the anew trained model showed the best results on the wiki data while the finetuned one dominated on the social media sample. It can be observed that the performance of the finetuned classifier directly correlates with the quality of the baseline model, which makes sense because the baseline serves its basis. However, it is not that a finetuned model improves by broadening the limitations of an initial model. It deals correctly with some part of the erroneous cases of the general classifier but makes mistakes in the other circumstances. Unfortunately, we did not manage to identify any specific patterns of the error in each experiment, it seems to be pretty random.

The classifiers attributed to the first 100 most-frequent tokens showed the highest scores overall almost in every experiment. The top 1000 and 1000+ groups seem to demonstrate similar performance. At this point, we must recall that the punctuation is included in the third group which means its performance is to some degree increased by the memorized thus always correctly predicted tokens. For a more fair comparison, the scores without considering the punctuation marks are provided in Table 5 - as they involve in the 1000+ group, only the performance of the model_other is affected.

dataset	experiment	with PUNCT	without PUNCT	dataset	experiment	with PUNCT	without PUNCT
main dev	baseline	**0.9813**	**0.9647**	wiki	baseline	0.9495	0.9287
	from scratch	0.9812	0.9629		from scratch	**0.9572**	**0.9395**
	finetuned	**0.9813**	**0.9647**		finetuned	0.9511	0.9309
main test	baseline	**0.9852**	**0.9653**	social	baseline	**0.919**	**0.8786**
	from scratch	0.9845	0.9633		from scratch	**0.919**	**0.8786**
	finetuned	**0.9852**	**0.9653**		finetuned	**0.919**	**0.8786**

Table 5. The scores of model_other with punctuation signs and without.

So, the performance of the model without punctuation tokens decreases by one to five hundredth depending on its percentage in evaluation data. Consequently, we can claim that in most cases it is the less frequent tokens that turn out to be the most problematic for morphological tagging, and most of them share ambiguity by grammatical features.

6.2. POS-tagging and morphological features assignment

The further section will refer to the analysis of part-of-speech tagging and grammatical features assignment separately. Tables 6 and 7 represent the scores of each classifier in POS-tagging while Tables 8, 9 demonstrate the performance on morphological features tagging. The scores were computed by comparing the corresponding parts of each tag, in case of incorrect POS assignment the grammatical features were still considered.

SynTagRus v2.3	experiment	model_top 100	model_top 1000	model_other	general
dev set	baseline	0.9855	**0.9889**	**0.9928**	**0.9904**
	from scratch	**0.9858**	0.9886	0.9924	0.9902
	finetuning	0.9855	**0.9889**	**0.9928**	**0.9904**
test set	baseline	**0.9885**	0.9896	**0.9920**	0.9908
	from scratch	0.9883	**0.9918**	0.9919	**0.9910**
	finetuning	**0.9885**	0.9896	**0.9920**	0.9908

Table 6. Scores of the systems on the main dataset in POS-tagging

dataset	experiment	model_top 100	model_top 1000	model_other	general
wiki	baseline	**0.9914**	0.9869	0.9847	0.9865
	from scratch	**0.9914**	**0.9935**	**0.9878**	**0.9894**
	finetuning	**0.9914**	0.9869	0.9847	0.9865
social media	baseline	**0.9441**	0.9423	**0.9806**	0.9640
	from scratch	0.9375	0.9423	0.9771	0.9601
	finetuning	0.9408	**0.9679**	**0.9806**	**0.9669**

Table 7. Scores of the systems on the additional datasets in POS-tagging

SynTagRus v2.3	experiment	model_top 100	model_top 1000	model_other	general
dev set	baseline	**0.9861**	**0.9809**	**0.9772**	**0.9800**
	from scratch	0.9858	0.9799	0.9762	0.9791
	finetuning	**0.9861**	**0.9809**	**0.9772**	**0.9800**
test set	baseline	**0.9886**	0.9817	**0.9766**	**0.9803**
	from scratch	0.9884	**0.9819**	0.9751	0.9794
	finetuning	**0.9886**	0.9817	**0.9766**	**0.9803**

Table 8. Scores of the systems on the main dataset in morphological features tagging

dataset	experiment	model_top 100	model_top 1000	model_other	general
wiki	baseline	0.9784	0.9542	0.9541	0.9596
	from scratch	0.9741	**0.9673**	**0.9602**	**0.9644**
	finetuning	**0.9828**	**0.9673**	0.9557	0.9634
social media	baseline	0.9178	0.8590	0.9208	0.9105
	from scratch	0.9112	**0.8654**	**0.9225**	0.9105
	finetuning	**0.9309**	**0.8654**	0.9208	**0.9154**

Table 9. Scores of the systems on the additional dataset in morphological features tagging

Concerning part-of-speech-tagging we mainly anticipated to gain a considerable improvement for the most-frequent units. As one can see, it did not happen. On the contrary, the quality of the baseline even deteriorates during the finetuning experiments with ensembles for the social media data. Instead, there is often an enhancement in the top1000 and 1000+ most frequent tokens in POS-tagging in comparison to the baseline which directly contrasts the predictions. However, we halfway met our expectations with regard to the grammatical features assignment. For the additional datasets, the anew trained and finetuned models gained a rather significant increase in quality for the less frequent words. This does not apply though to the main sample where the baseline model almost everywhere surpasses the produced ensembles.

Interestingly, in most cases it is one and the same classifier that performs the best for both part-of-speech and grammatical features assignment in general. It could indicate that these essences are highly related for a model during morphological tagging, so any attempt to somehow split them would not lead to an improvement. This tendency, nonetheless, is not shared across the groups of tokens: usually the best in general classifier does not obtain the highest score on every single frequency group. Thus, if combining the best-performance models on each set of tokens, we would get an ensemble surpassing all the regarded systems apart from the development set of the main sample, so one of the main goals of this research could be considered achieved to some extent.

6.3. Discussion

Table 10 presents the best possible ensemble of the classifiers for each dataset and its score.

dataset	top100	top1000	1000+	general
main dev	baseline	baseline	baseline	0.9774 → 0.9774
	0.9813	0.9783	0.9756	
main test	baseline	from scratch	baseline	0.9783 → 0.9784
	0.9852	0.9799	0.9752	
wiki	finetuned	from scratch	from scratch	0.9538 → 0.9625
	0.9741	0.9672	0.9572	
social media	finetuned	finetuned	baseline	0.9027 → 0.9105
	0.9211	0.8590	0.919	

Table 10. The score of the best possible ensemble of classifiers.

We managed to improve significantly on two additional datasets. There was also a slight increase in the score on the test set of the main sample but no enhancement in the development data. Since the social media and wiki datasets differ from the main sample in size, and one could come up with an explanation of the size being impactful on the results of such a proposed approach, we can not claim that we have achieved the improvement of quality by only integrating the knowledge of the homonymy types.

During the review of erroneous cases, we tried to discover some patterns indicating that the model has learned to better deal with a certain type of ambiguity, but we failed. In most cases, the provided ensembles make the same mistakes as a baseline; the majority of such errors are outlined in the GramEval 2020 report (Lyashevskaya et al., 2020, p. 12). We found no instances of the complete overcoming of these limitations. There are still some specific tendencies inherent in the anew trained and finetuned classifiers' predictions but they could mostly be explained by other contributing factors. For instance, the from scratch model attributed to the group of most frequent tokens is inclined to produce a wrong gender assignment even for the not controversial cases - it can happen due to the lack of training data as most tokens in this group do not share the grammatical feature of gender. This problem

does not arise for the baseline classifier which thus learns how to correctly assign the gender tag. Still, such interpretation has nothing to do with homonymy types.

In general, there are multiple factors indicating the failure of the proposed approach despite the fact of the increase in a score. Such is the inconsistency across different datasets, rather controversial results in part-of-speech and grammatical features tagging, and lack of evidence revealed in error analysis confirming that such method clearly impacts the results.

7. Conclusions

To summarize, this paper is devoted to one of the first NLP tasks - full morphological tagging. Despite some of the latter approaches achieving an upper-tier score, this topic still merits attention, as morphological parsing is often conducted at the early stage of many NLP pipelines, thus, an error that occurred at this point would propagate further and might result in deterioration of the model performance. In this research, we expected to provide an algorithm that would improve the performance of an existing approach to grammatical parsing. We attempted to benefit from linguistic knowledge, concerning different types of homonymy and their correlation with tokens' frequencies. Our main hypothesis lies in the fact that we could increase the quality of morphological tagging by training different classifiers on tokens, sharing different ambiguation properties, and then applying each classifier to the appropriate group of tokens it was trained on.

As a foundation for our classifiers, we utilized the architecture of morpho_ru_syntagrus_bert - advanced BERT model from the DeepPavlov library. At first, we split the words from the training sample by their frequency into three groups, each is believed to contain tokens with similar ambiguity properties; then we trained separate models on a certain group. During the prediction for every token, a corresponding classifier was used. We tested our ensemble on four datasets of different sizes: the SynTagRus v2.3 as the main one, and the collection from GramEval 2020 data.

The results of applying the proposed technique turned out rather controversial. On the one hand, we managed to gain an improvement straight away on wiki and social media dataset, and by combining our classifiers with the baseline we could slightly increase the score on the test set of the main corpora as well. On the other hand, we can not consider the enhancement significant enough to make a claim. Moreover, the purpose of this research was not the pursuit of high-score performance, but an attempt to test the invented technique that allegedly integrates the linguistic knowledge of different homonymy types. Unfortunately, we

could not come up with clear evidence of it: there have been no characteristic patterns revealed that would indicate a certain model learning to better deal with an attributed ambiguity. On the contrary, we managed to discover some other factors that are likely to cause such behavior.

One of the unstated prerequisites for this research was the prohibition on using third-party resources in a produced system. That is why we chose to split tokens by their frequencies assuming the correlation between the homonymy types and word frequency distribution: we do not need any additional means for such partition. However, there are no direct dependencies between these essences, so each produced group contained quite a mix of tokens with diverse ambiguity properties. That would be of interest to examine the performance of a similar system but with a split done straightforwardly by homonymy types. For that to happen one would need to utilize some dictionary resources containing information on all the possible grammatical analyses for every token, for instance, pymorphy2. Perhaps such a technique would lead to a better performance than the one that we managed to achieve in this research.

References

Anastasyev D. (2020). Exploring pretrained models for joint morpho-syntactic parsing for Russian. In Computational Linguistics and Intellectual Technologies. Proceedings of the International Conference Dialogue 2020, Vol. 19.

Anastasyev D., Andrianov A., Indenbom E. (2017). Part-of-speech Tagging with Rich Language Description.

Anastasyev D., Gusev I., Indenbom E. (2018), Improving Part-of-Speech Tagging via Multi-task Learning and Character-level Word Representations, Computational linguistics and intellectual technologies: Proceedings of the International Conference “Dialog 2018”, pp. 14–27.

Awwalu, J., Abdullahi, S. E.-Y., & Evwiekpaefe, A. E. (2020). Parts of speech tagging: A review of techniques. FUDMA JOURNAL OF SCIENCES, 4(2), 712–721. https://doi.org/10.33003/fjs-2020-0402-325

Brill, E. (1992). A Simple Rule-Based Part of Speech Tagger. ANLP, 152–155.

Dereza O., Kayutenko D., Fenogenova A. (2016). AUTOMATIC MORPHOLOGICAL ANALYSIS FOR RUSSIAN: A COMPARATIVE STUDY. National Research University Higher School of Economics, Moscow, Russia.

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

Garside R. (1987). The CLAWS word-tagging system. The Computational analysis of English: A corpus-based approach. London: Longman. . C. 30-41.

Gimenez J., Marquez L. (2003). Fast and Accurate Part-of-Speech Tagging: The SVM Approach Revisited.TALP Research Center, LSI Department Universitat Politecnica de Catalunya Jordi Girona Salgado 1{3, E-08034, Barcelona

Greene B.B., Rubin G.M. (1971). Automatic grammatical tagging of English // Department of Linguistics, Brown University.

Kanerva J., Ginter F., Miekka N., Leino A., Salakoski T. 2018. Turku Neural Parser Pipeline: An End-to-End System for the CoNLL 2018 Shared Task. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 133–142, Brussels, Belgium. Association for Computational Linguistics.

Klyshinsky E.S., Logacheva V.K., Mansurova O.YU., Maksimov V.YU., Karpik O.V., Ziyaztdinov I.B., Makeenko P.A. (2015). Issledovanie neodnoznachnosti upotrebleniya slov v evropejskih yazykah // Preprint IPM im. M.V.Keldysha.. No 4. 31 s. URL:

Klyshinsky E.S., Logacheva V.K., Karpik O.V., Bondarenko A.V. (2020) Kolichestvennaya ocenka grammaticheskoj neodnoznachnosti nekotoryh evropejskih yazykov // Vestnik Novosibirskogo gosudarstvennogo universiteta. Lingvistika i mezhkul'turnaya kommunikaciya. 18(1). S. 5-21.

Klyshinsky E.S., Buntyakova V.A., Karpik O.V., (2021) Issledovanie grammaticheskoj neodnoznachnosti naibolee chastotnyh slov russkogo yazyka. Preprint IPM im. M.V.Keldysha.

Kucera H., Francis W.N. (1967). Computational analysis of present-day American English. University Press of New England. . 424 c.

Kumawat D., Jain, V. (2015). POS Tagging Approaches: A Comparison. International Journal of Computer Applications, 118(6), 975–8887. Retrieved from

Lyashevskaya O., Shavrina T., Trofimov I., Vlasova N. (2020). GRAMEVAL 2020 SHARED TASK: RUSSIAN FULL MORPHOLOGY AND UNIVERSAL DEPENDENCIES PARSING.

Segalovich, I. (2003).: A fast morphological algorithm with unknown word guessing induced by a dictionary for a web search engine. In: Proceedings of MLMTA-2003, pp. 273–280

Sorokin A., Shavrina T., Lyashevskaya O., Bocharov V., Alexeeva S., Droganova K., Fenogenova A., Granovsky D. (2017). MorphoRuEval-2017: an Evaluation Track for the Automatic Morphological Analysis Methods for Russian.

Sokirko, A.V. (2004). Morphologicheskie moduli na sajte www.aot.ru. In Proc. DIALOG'04. In Russian.

Sharoff, S., Kopotev, M., Erjavec, T., Feldman, A., Divjak, D. (2008).: Designing and evaluating Russian tagsets. In: Proceedings of LREC-2008, pp. 279–285, Marrakech

Schmid H. (1994). Probabilistic Part-of-Speech Tagging Using Decision Trees. Institut für maschinelle Sprachverarbeitung, Universität Stuttgart.

Thede S. and Harper M. (1999). A Second-Order Hidden Markov Model for Part-of-Speech Tagging. In Proceedings of the 37th Annual Meeting of the ACL.

Qi P., Dozat T., Zhang Y., Manning C. 2018. Universal Dependency Parsing from Scratch. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 160–170, Brussels, Belgium. Association for Computational Linguistics.