Exploring the Limits of Word Sense Disambiguation for Russian using Automatically Labelled Collections

Angelina Bolshina[a], Natalia Loukachevitch[a]

[a] *Lomonosov Moscow State University, GSP-1, Leninskie Gory, Moscow, 119991, Russian Federation*

Abstract

There is a long-standing problem in the field of the word sense disambiguation (WSD) that is known as the knowledge acquisition bottleneck. Many state-of-the-art WSD algorithms are data-hungry, so the lack of the sense-annotated data hinders the development of supervised WSD models for the low-resource languages such as Russian. In this work we introduce an algorithm of automatic generation and labelling of the training collections based on the monosemous relatives concept. This method relies on the RuWordNet thesaurus and the relations between ambiguous words and the monosemous words they are connected to. Our approach addresses the issues of the limited availability of the examples for some polysemous word senses and the bias that can be possibly introduced by some training samples. The experiments attested that the generated collections enable a wide coverage of the polysemous words presented in RuWordNet, and the models trained on these collections can attain a good overall performance on the Russian WSD benchmarks.

Keywords

Word sense disambiguation, knowledge acquisition bottleneck, Russian dataset, monosemous relatives, ELMo, BERT

1. Introduction

Word sense disambiguation (WSD) is a fundamental task in computational lexical semantics that is aimed at predicting the correct sense of a polysemous word in a given context from a predefined sense inventory. WSD is widely used in many semantic-oriented applications such as semantic role labelling, knowledge graph construction, machine translation, question answering, and entity linking, etc. WSD is a supervised task, and this implies that sophisticated models, which can attain the competitive results, require a large amount of labelled data. Expert annotation of datasets for this task is rather expensive in terms of time and money, and large hand-crafted corpora with the sense annotation can be found mostly for English [1, 2]. The restricted availability of sense-tagged data does not allow to scale existing WSD systems across many languages. For the Russian language there exist only several small datasets with sense labels, however, it is not enough for training any state-of-the-art model.

One of the possible alternatives to manual annotation is an automatic acquisition of training samples. In our research we investigate the method to automatically generate and label training collections with the help of monosemous relatives, that is a set of unambiguous words (or phrases) related to particular senses of a polysemous word. However, as it was noted in [3], some senses of target words do not have monosemous relatives, and the noise can be introduced by some distant relatives. In our research we tried to address these issues.

The main contribution of this study is that we have expanded a set of monosemous relatives under consideration via various semantic relations and distances: in comparison with earlier approaches, now monosemous relatives can be situated at a greater distance from a target ambiguous word in a graph. Moreover, we have introduced a numerical estimation of a similarity between a monosemous relative and a particular sense of a target word which is further used in the development of the training

Proceedings of the Linguistic Forum 2020: Language and Artificial Intelligence, November 12-14, 2020, Moscow, Russia
EMAIL: louk_nat@mail.ru (2)
ORCID: 0000-0002-9106-7192 (1); 0000-0002-1883-4121 (2)

collection. To evaluate the created training collections, we utilized contextualized word representations – ELMo [4] and BERT [5]. We also explored the ways of augmenting automatically generated collections with the manually labelled samples. The source code of our algorithm and experiments is publicly available at: https://github.com/loenmac/russian_wsd_data.

The paper is organized as follows. In section two we review the related work. Section three is devoted to the data description. The fourth section describes the method applied to automatically generate and annotate training collections. The procedure of creating the collections is explained in the fifth section. In the sixth section, we describe a supervised word sense disambiguation algorithm trained on our collected material and demonstrate the results obtained by four different models. In this section we also present a comparative analysis of the models trained on different kinds of train collections. Concluding remarks are provided in the seventh section.

2. Related Work

To overcome the limitations, that are caused by the lack of annotated data, several methods of generating and harvesting large train sets have been developed. There exist many techniques based on different kinds of replacements, which do not require human resources for tagging. The most popular method is that of monosemous relatives [6]. Usually, WordNet [7] is used as a source for such relatives. WordNet is a lexical-semantic resource for the English language that contains a description of nouns, verbs, adjectives, and adverbs in the form of semantic graphs. All words in those networks are grouped into sets of synonyms that are called synsets.

Monosemous relatives are those words or collocations that are related to the target ambiguous word through some connection in WordNet, but they have only one sense, i.e. belong only to one synset. Usually, synonyms are selected as relatives but in some works hypernyms and hyponyms are chosen [8]. Some researchers replace the target word with named entities [9], some researchers substitute it with meronyms and holonyms [10]. In the work [3] distant relatives (including distant hypernyms and hyponyms) were used; the procedure of training contexts selection was based on the distance to a target word and the type of the relation connecting the target sense and a monosemous relative.

In the article [11] a special algorithm was created in order to select the best replacement out of all words contained within synsets of the target word and neighboring synsets. The algorithm described in [12] to construct an annotated training set is a combination of different approaches: monosemous relatives, glosses, and bootstrapping. Monosemous relatives can be also used in other tasks, for example, for finding the most frequent word senses in Russian [13]. Other methods of automatic generation of training collections for WSD exploit parallel corpora [2], Wikipedia and Wiktionary [14, 15], topic signatures [16]. [17] created large training corpora exploiting a graph-based method that took an unannotated corpus and a semantic network as an input. Algorithm MuLaN (Multilingual Label propagatioN) is based on the label propagation [18]. In this novel method, the authors utilize contextualized word embeddings, information from a knowledge base and projection of the sense tags from a high-resource language to a low-resource one. A profound survey on various manual, semi-automatic and automatic approaches to tackle the issue of knowledge acquisition bottleneck is provided in [19].

Various supervised methods including kNN, Naive Bayes, SVM, neural networks were applied to word sense disambiguation [20]. Contextualized embeddings, like BERT [5], ELMo [4], and context2vec [21], have also proven to be suitable for the WSD task: [22, 23, 24, 25]. The most widely used deep contextualized embeddings are ELMo [4] and BERT [5].

In ELMo (Embeddings from language models) [4] context vectors are computed in an unsupervised way by two layers of bidirectional LSTM, that take character embeddings from convolutional layer as an input. Character-based token representations help to tackle the problems with out-of-vocabulary words and rich morphology. BERT (Bidirectional Encoder Representations from Transformers) [5] has a different type of architecture, namely multi-layer bidirectional Transformer encoder. During the pre-training procedure, the model is “jointly conditioning on both left and right context in all layers” [5]. Moreover, BERT uses WordPiece tokens, that is subword units of words, which also helps to avoid the problem of out-of-vocabulary words. Since these contextualized word embeddings imply capturing polysemy better than any other representations and, thus, we employ them in our investigation.

3. Data

In our research as an underlying semantic network, we exploit Russian thesaurus RuWordNet [26]. It is a semantic network for Russian that has a WordNet-like structure. In total it contains 111.5 thousand of words and word combinations for the Russian language. RuWordNet was used to extract semantic relations (e.g. synonymy, hyponymy etc.) between a target sense of a polysemous word and all the words (or phrases) connected to it, including those linked via distant paths. The sense inventory was also taken from this resource. RuWordNet contains 29297 synsets for nouns, 63014 monosemous and 5892 polysemous nouns. In this research we consider only ambiguous nouns. Table 1 presents a summary of the number of senses per noun:

Table 1

Quantitative characteristics of polysemous words in RuWordNet

Number of senses of a polysemous word	Number of words in RuWordNet
2 senses	4271
3 senses	997
4 senses	399
5 senses	149
> 5 senses	76
Total number of senses	14 357

We utilized two corpora in the research. A news corpus consists of news articles harvested from various news sources. The texts have been cleaned from HTML-elements or any markup. Another corpus is Proza.ru, a segment of Taiga corpus [27], which is compiled of works of prose fiction. We exploit these two corpora in to compare the performance of the WSD models trained on the collections obtained with these resources.

Table 2

Cases when a word from the RUSSE’18 dataset was not included in the final test set

Explanation	Number of words	Example
A word has only one meaning in RuWordNet	34	The word *двойник* ‘doppelganger’ has only one meaning in RuWordNet whereas in RUSSE’18 it has 4.
A word is missing in the RuWordNet vocabulary	9	The word *гипербола* ‘hyperbole’.
The senses from RuWordNet and RUSSE’18 dataset have only one sense in common	4	The word *мандарин* has two senses described in RUSSE’18: its sense ‘tangerine’ is included in the thesaurus, whereas its meaning ‘mandarin, bureaucrat’ is absent.
Controversial cases of sense mapping	29	The word *демократ* ‘democrat’ has 2 senses: ‘supporter of democracy’ and ‘a member of the Democratic Party’. But there’s another one in RUSSE’18: ‘a person of a democratic way of life, views’.
Not enough examples for senses in the corpora	2	Words *карьер* ‘quarry/a very fast gallop’ and *шах* ‘shah/check’.
Words with morphological homonymy	1	The word *суда* ‘court (Gen, Sg)/ship (Nom, Pl)’. Those words have distinct lemmas.

For evaluation of our algorithm of training data generation, we used three distinct RUSSE'18 datasets for Russian [28]. These datasets were created for the shared task on word sense induction for the Russian language. The first dataset is compiled from the contexts of the Russian National Corpus. The second dataset consists of the contexts from Wikipedia articles. And the last dataset is based on the Active Dictionary of the Russian Language [29] and contains contexts taken from the examples and illustration sections from this dictionary. All the polysemous words are nouns. From the RUSSE dataset, we excluded some polysemous words, and in Table 2 we overview the common reasons why it was done.

The final list of the target ambiguous words contains 30 words in total, each having two different senses. We will call the resulting test dataset RUSSE-RuWordNet because it is a projection of RUSSE'18 sense inventory on the RuWordNet data.

We also created a small training dataset, that consists of the word sense definitions and examples of uses from Ozhegov dictionary [30] for every target polysemous word. This training data is utilized as a baseline for the WSD task. In this set each sense of ambiguous word has one definition and between 1 and 3 usage examples.

Table 3 demonstrates quantitative characteristics of all of the above-mentioned corpora.

Table 3
Quantitative characteristics of the corpora and datasets used in the experiments

	Taiga-Proza.ru	**News Corpus**	**RUSSE-RuWordNet**	**Dictionary Corpus (Baseline)**
Number of sentences	32,8 million	24,2 million	2 103	144
Number of lemmas	246,8 million	288,1 million	39 311	657
Number of unique lemmas	2,1 million	1,4 million	12 110	475

4. Candidate Selection and Ranking Algorithm

The underlying concept of our algorithm is a concept of monosemous relatives, that is a set of unambiguous words (or phrases) related to a particular sense of a polysemous word. Our approach for collecting a training corpus is based on the substitution: for every polysemous word we select appropriate monosemous relatives, then in a text, the occurrences of these relatives are substituted by the target polysemous word and these instances are labelled with a sense tag of a monosemous relative.

A central part of our method belongs to the candidate selection and ranking algorithm. Not all monosemous relatives can serve as a representation of a target word sense, that is why we developed a system that assigns a weight to every candidate monosemous relative, and based on this score a ranked list of all possible candidates is constructed. Moreover, this algorithm helps to verify the usage of a monosemous relative in a corpus, because some words marked as monosemous in the thesaurus may have more than one sense in a corpus.

To extract the features necessary for computing candidate weights, we utilize RuWordNet thesaurus. The nodes of this semantic graph are represented as groups of synonyms, called synsets, and the edges are relations between these groups of words.

When constructing a training set, we take into account not only the close relations like synonymy, hypernymy and hyponymy, but also far more distant ones, for example, co-hyponymy. Our findings from the previous research [31] prove, that the inclusion of the words connected to a target ambiguous word via distant relations does not have a negative effect on the performance of the WSD model. Moreover, the utilization of such distant relatives enables a wider coverage of the polysemous words from the thesaurus in a training collection. In our research, the distance between the target sense of the polysemous word and its candidate monosemous relatives can reach up to 4 steps in the semantic graph. The final list of monosemous relatives, which will be exploited in the training collection, is composed of the candidate monosemous relatives selected during ranking procedure. Candidate monosemous relatives are unambiguous words and phrases, that can be located in up to four-step relation paths to a polysemous word and include co-hyponyms, two-step (or more) hyponyms and hypernyms, and the weights of these monosemous relatives are yet to be estimated.

Another constituent of our system is the notion of ***a synset nest***. The synset nest represents a set of words (or phrases) most closely related to a particular sense of the target word, specifically target word synonyms and all the words from directly related synsets within 2 steps from the target word. We use this set of words when computing a score for a candidate monosemous relative in order to identify how similar is the sense of the candidate to the sense of the target polysemous word. A fragment of the nest for the word *такса* ‘dachshund’ is given below:

1) “*охотничий пёс, охотничья собака, пёсик, четвероногий друг, псина, собака, терьер, собачонка, борзая собака…*” / ‘hunting dog, hunting dog, doggie, four-legged friend, dog, dog, terrier, dog, greyhound dog…’

In order to ensure, that the samples with monosemous relatives extracted from a corpus will serve as a good representation of the target sense, we employ in our candidate selection and ranking algorithm a custom word2vec embedding model trained on the same corpus from which the contexts are retrieved. In this work we utilized word2vec embedding models [32] based on neural network architecture CBOW.

Our selection and ranking method, thus, consists of the following steps:

1. We extract all the candidate monosemous relatives within 4 steps from a target polysemous word sense $\boldsymbol{s_j}$.
2. We compile the nest $\boldsymbol{ns_j}$ which consists of synonyms to a target sense and all the words from the synsets within 2 steps from a target word $\boldsymbol{s_j}$. The nest $\boldsymbol{ns_j}$ consists of $\boldsymbol{N_k}$ synsets.
3. For each candidate monosemous relative $\boldsymbol{r_j}$, we find the most similar words according to the word2vec model trained on a reference corpus.
4. We intersect this list of similar words with the words included in the nest $\boldsymbol{ns_j}$ of the target sense $\boldsymbol{s_j}$.
5. For each word in the intersection, we take its cosine similarity weight calculated with the word2vec model and assign it to the synset it belongs to. The final weight of the synset in the nest $\boldsymbol{ns_j}$ is determined by the maximum weight among the words $\boldsymbol{w_{k_1}^{j}}, \dots, \boldsymbol{w_{k_i}^{j}}$ representing this synset in the intersection.
6. The total score of the monosemous candidate $\boldsymbol{r_j}$ is the sum of the weights of all synsets from the nest $\boldsymbol{ns_j}$. Thus, the final weight of the candidate can be defined as follows:

$$Weight_{r_j} = \sum_{k=1}^{N_k} \max\left[cos\left(r_j,\ w_{k_1}^{j}\right), \dots, cos\left(r_j, w_{k_i}^{j}\right)\right] \qquad (1)$$

The formula was designed to assign higher scores to those candidates, that resemble a greater number of synsets from the nest close to the target sense of the ambiguous target word. For example, these are the monosemous relatives ratings for the two senses of the word *абрикос* ‘apricot’ (relatives weights are given in brackets):

2) “Tree”: *яблоня* ‘apple tree’ (6.3), *яблонька* ‘small apple tree’ (4.9), *олива* ‘olive tree’ (4.8), *смоковница* ‘fig tree’ (3.3), *терновник* ‘blackthorn’ (3.0), *плодовое дерево* ‘fruit tree’ (2.9) … etc.

3) “Fruit”: *инжир* ‘fig’ (6.8), *яблоко* ‘apple’ (6.4), *смоква* ‘fig’ (6.0), *ранет* ‘variety of small apples’ (5.7), *антоновка* ‘variety of apples’ (4.9), *фрукт* ‘fruit’ (4.3) … etc.

These examples demonstrate that different sets of monosemous relatives can help to distinguish between the senses of a target polysemous word. The scores assigned to the monosemous relatives are not absolute, the range of the score values usually depends on the number of the monosemous candidates. For example, the word *лицо* ‘person’ has around 2000 candidate monosemous relatives and the highest score among them is 24, the word *идея* ‘concept’ has 8 candidates with 2.3 being the highest score, and the word *рулет* ‘meatloaf’ has only one monosemous relative and its weight is 0.5.

To estimate how many polysemous word senses from RuWordNet our method can cover, we found candidate monosemous relatives for the ambiguous nouns in the thesaurus using our algorithm but without word2vec filter. Only two words out of 5895 do not have monosemous relatives within the four-step relation path in the RuWordNet graph.

5. Generating Training Data using Monosemous Relatives

For comparison, we decided to create two separate training collections compiled from the news and Proza.ru corpora, and we also exploited two distinct approaches to a collection generation. In Table 4 we present the quantitative characteristics of the two collections, such as the relations connecting the target senses and their monosemous relatives, distances between them, and a proportion of monosemous relatives expressed as a phrase.

The first collection was compiled only with a monosemous relative from the top of the candidate rating. We wanted to obtain 1000 examples for each of the target words, but sometimes it was not possible to extract so many contexts with one particular candidate. That is why in some cases we also took examples with words next on the candidates' list. For simplicity, we call this collection Corpus-1000 because we obtained exactly 1000 examples for each sense.

The second approach enables to harvest more representative collection with regard to the variety of contexts. The training examples for the target ambiguous words were collected with the help of all respective unambiguous relatives with non-zero weight. The number of extracted contexts per a monosemous candidate is in direct proportion to its weight. We name this collection a balanced one because the selection of training examples was not restricted to the contexts which have only one particular monosemous relative.

Table 4
Quantitative characteristics of monosemous relatives included in the balanced training collection.

Distance to a target sense	Proportion of occurrences in the news collection	Proportion of occurrences in Proza.ru collection
0 (synset)	2%	4%
1	13%	9%
2	38%	37%
3	31%	34%
4	16%	16%
Relation between a target sense and a monosemous relative		
Synonyms	2%	4%
Hyponyms	13%	8%
Hypernyms	11%	9%
Cohyponyms	28%	28%
Cohyponyms situated at three-step path	24%	28%
Cohyponyms situated at four-step path	19%	22%
Other	3%	1%
Word combinations	48%	29%

Two word2vec embedding models that we used in our experiments were trained separately on the news and Proza.ru corpora with the window size of 3. As a preprocessing step, we split the corpora into separate sentences, tokenized them, removed all the stop words, and lemmatized the words with pymorphy2 tool [33]. For each candidate monosemous relative with the help of these models, we extracted 100 most similar words, that are used to find an intersection with a synset nest. The words obtained from the word2vec models were filtered out – we removed the ones not included in the thesaurus.

6. Experiments

We conducted several experiments with the generated text collections to evaluate the quality of the disambiguation, which can be achieved using them. Following [23], in our research we used an easily interpretable classification algorithm – non-parametric nearest neighbor classification (kNN) based on the contextualized word embeddings ELMo and BERT. Contextualized embeddings derived for the training data form the clusters in the vector space, then for each test sample representation we find *k* closest training examples in the feature space, and according to the class of these neighbors we define the output sense of the test sample.

In our experiments we exploited two distinct ELMo models – the one trained by DeepPavlov on Russian WMT News and the other is RusVectōrēs [34] lemmatized ELMo model trained on Taiga Corpus [27]. These models can be used in two ways: we can extract a vector for a whole sentence with a target word, and also just a single vector for a target ambiguous word can be obtained. We also used two BERT models: BERT-base-multilingual-cased released by Google Research and RuBERT, which was trained on the Russian part of Wikipedia and news data by DeepPavlov [35]. To extract BERT contextual representations, we followed the method described by [5] and [23] and concatenated "the token representations from the top four hidden layers of the pre-trained Transformer" [5].

Table 5
F1 scores for BERT-based WSD models

Model	RuBERT DeepPavlov (Corpus-1000 collection)		Multilingual BERT (Corpus-1000 collection)		RuBERT DeepPavlov (balanced collection)		Multilingual BERT (balanced collection)	
k	Proza.ru	News collection	Proza.ru	News collection	Proza.ru	News collection	Proza.ru	News collection
5	0.793	0.771	0.694	0.667	0.792	0.769	0.717	0.682
7	**0.804**	**0.774**	0.699	0.673	0.802	0.768	0.723	0.683
9	0.802	0.769	**0.7**	**0.677**	**0.812**	**0.774**	**0.729**	**0.688**
Baseline	0.667		0.672		0.667		0.672	

Table 6
F1 scores for ELMo-based WSD models

Model	ELMo RusVectōrēs (target word, Corpus-1000)		ELMo DeepPavlov (whole sentence, Corpus-1000)		ELMo RusVectōrēs (target word, balanced collection)		ELMo DeepPavlov (whole sentence, balanced collection)	
k	Proza.ru	News collection	Proza.ru	News collection	Proza.ru	News collection	Proza.ru	News collection
1	0.809	0.794	0.765	**0.752**	0.812	0.797	0.745	0.758
3	0.826	0.811	**0.773**	0.749	0.833	0.81	0.775	0.753
5	0.834	**0.819**	0.77	0.748	0.845	0.81	0.776	0.756
7	**0.841**	**0.819**	0.767	0.746	**0.857**	0.815	**0.793**	**0.759**
9	0.84	0.816	0.762	0.747	0.856	**0.821**	0.791	0.753
Baseline	0.772		0.716		0.772		0.716	

Table 7
F1 scores for ELMo-based WSD models: Proza.ru, balanced collection

Model	ELMo RusVectōrēs **(whole sentence)**	ELMo DeepPavlov **(target word)**	ELMo-ruwikiruscorpora (non-lemmatized, target word)
k			
1	0.807	0.723	0.776
3	0.824	0.73	**0.794**
5	**0.827**	0.738	0.792
7	0.824	0.736	0.792
9	0.821	**0.742**	**0.794**
Baseline	0.772	0.716	-

Table 8
F1 scores for ELMo RusVectōrēs WSD models: Proza.ru and News balanced collections augmented with dictionary definitions

Model	ELMo RusVectōrēs (target word)	ELMo RusVectōrēs (target word)
k	Proza.ru	News collection
1	0.819	0.824
3	0.835	0.832
5	0.847	0.828
7	**0.859**	0.834
9	0.858	**0.842**

Table 5 and Table 6 demonstrate the results obtained by different types of contextualized word embeddings, the training collections, and model parameters. As it can be seen, all the systems surpassed the quality level of the baseline solution trained on the dataset of the dictionary definitions and usage examples.

The algorithm based on the ELMo pre-trained embeddings by RusVectōrēs outperformed all other models and achieved 0.857 F1 score. The second-best model in the WSD task is RuBERT by DeepPavlov, followed by ELMo model by DeepPavlov. The lowest F1 score belongs to Multilingual BERT.

As for the difference in F1 scores between the Corpus-1000 and the balanced collection, we can observe the performance drop for the Corpus-1000 for all the models, which means that the approach used to generate the balanced collection is better suited for the task. Corpus-1000 does not include all possible monosemous relatives, so the collection lacks contextual diversity, the balanced collection, on the contrary, is more representative with regard to the variety of contexts.

The Proza.ru model achieves better results and outperforms the news model. The qualitative analysis of the classification errors caused by the model trained on the news collection showed that the main cause of mistakes were lexical and structural differences between training and test sets.

As we have already mentioned, ELMo contextualized embeddings can be exploited in two different ways, and in our research, we wanted to explore, which one is best suited for the task and the models. The first two columns of Table 7 demonstrate the results of classification on the RusVectōrēs and DeepPavlov ELMo embeddings extracted differently from the cases described in Table 6. It can be seen, that these modes of use led to the lower F1 score for both of the models. Thus, the optimal way to use RusVectōrēs ELMo embeddings for the WSD task with kNN-classifier is to extract embedding solely for a target polysemous word, whereas for the DeepPavlov ELMo model it is recommended to extract the representation for the whole sentence with the polysemous word.

The results of the research [22] showed that lemmatized training data can improve ELMo performance in word sense disambiguation for the Russian language. In our study we wanted to prove that this also holds true for our automatically generated training collections. We compared two RusVectōrēs ELMo models: lemmatized model trained on Taiga and token-based model trained on the Russian Wikipedia and the Russian National Corpus. As the training collection, we used Proza.ru (balanced) in two variants – lemmatized and simply tokenized. The results for non-lemmatized input are presented in the last column of Table 7. It turns out that even for the generated training collections ELMo model on lemmas outperforms ELMo on tokens. Thus, lemmatized input to the WSD models is preferable for the Russian language as it does not contain any additional morphological information, which is excessive for the lexical-semantic task.

Another experiment was aimed at the evaluation of the models trained on the automatically generated collections augmented with the dictionary definitions from the corpus used in the baseline solution. Since the very first works in the field of WSD [36], glosses have proven to be a valuable source of information, and nowadays word definitions are also incorporated in the models: [37, 38, 39, 40]. The outline was as follows: we enriched Proza.ru and News balanced collections with manually annotated dictionary definitions and examples of use and then applied kNN-classifier to the contextualized embeddings extracted for this augmented collection (ELMo RusVectōrēs). The results are presented in Table 8.

Even though the number of additional examples is rather small, we can still see some minor improvements in the performance of the Proza.ru collection and a 2% increase in the F1-score of the News model.

In our recent work [41], we compared the WSD model performance trained on the automatically and manually labelled data. In this case we also used RusVectōrēs ELMo contextualized embeddings as they show the best quality in all the settings. We made 5 random divisions of RUSSE-RuWordNet dataset into train and test sets in the ratio 2:1. Then we used this data to train and test 5 different WSD models. The 5-fold cross-validation in this setup amounted to 0.917 F1. Then we combined our news training collection with each train set described above, and measured the performance on the corresponding test sets. The F1 score was 0.94.

This experiment demonstrated that the WSD model trained on the automatically labelled data gives the results comparable with the results obtained with the hand-labelled data. Moreover, the metrics obtained in these experiments show that manually labelled data combined with the generated one can boost the overall performance. Among all the possible ways of augmenting a training collection with manually-curated samples, the data from lexical resources, such as dictionaries, seems to be the most convenient as it is easy to be obtained.

To explore how contextualized embeddings from the training, test and baseline collections are located relative to each other in a vector space, we visualized them with t-SNE algorithm. The contextualized representations were extracted from RusVectōrēs ELMo model.

Figure 1 and Figure 2 demonstrate that all of the samples of the same sense occupy similar parts of the vector space. The examples from the dictionary corpus are situated near the border of the sense clusters both in the News and Proza.ru collections representations. But this configuration is not characteristic for every target polysemous word: in some cases, sense groupings from the different datasets occupy distinct parts of the vector space or some words may not have such clear-cut sense groupings. Such representation, for example, was obtained for the word *слог* as depicted on Figure 3. We noticed, that the diversity in polysemous words representations has a direct correspondence with the F1-score obtained for the target senses: for example, the F1 for the word *крона* equals to 0.93, whereas the same metric for the word *слог* is only 0.62.

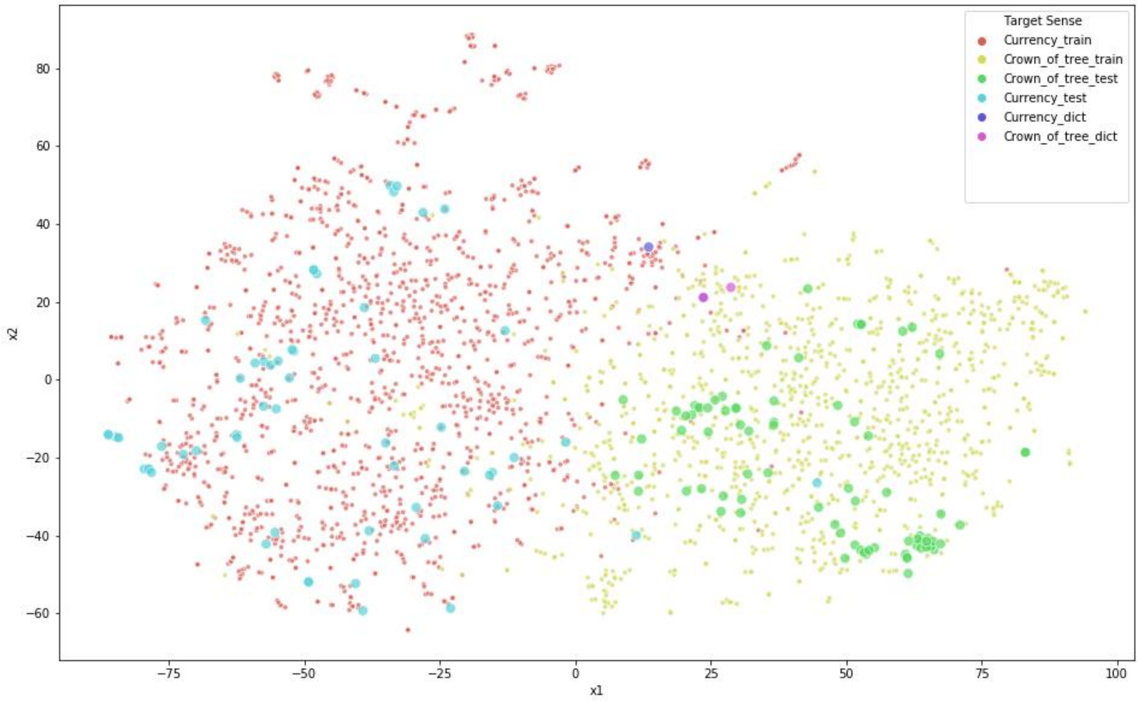

Figure 1: Representations for the word *крона* encoded by RusVectōrēs ELMo model, samples marked with “_train” label are taken from the News train collection (balanced), examples marked with “_test” are taken from the manually annotated evaluation collection contexts, label “_dict” stands for the examples of use or dictionary definitions

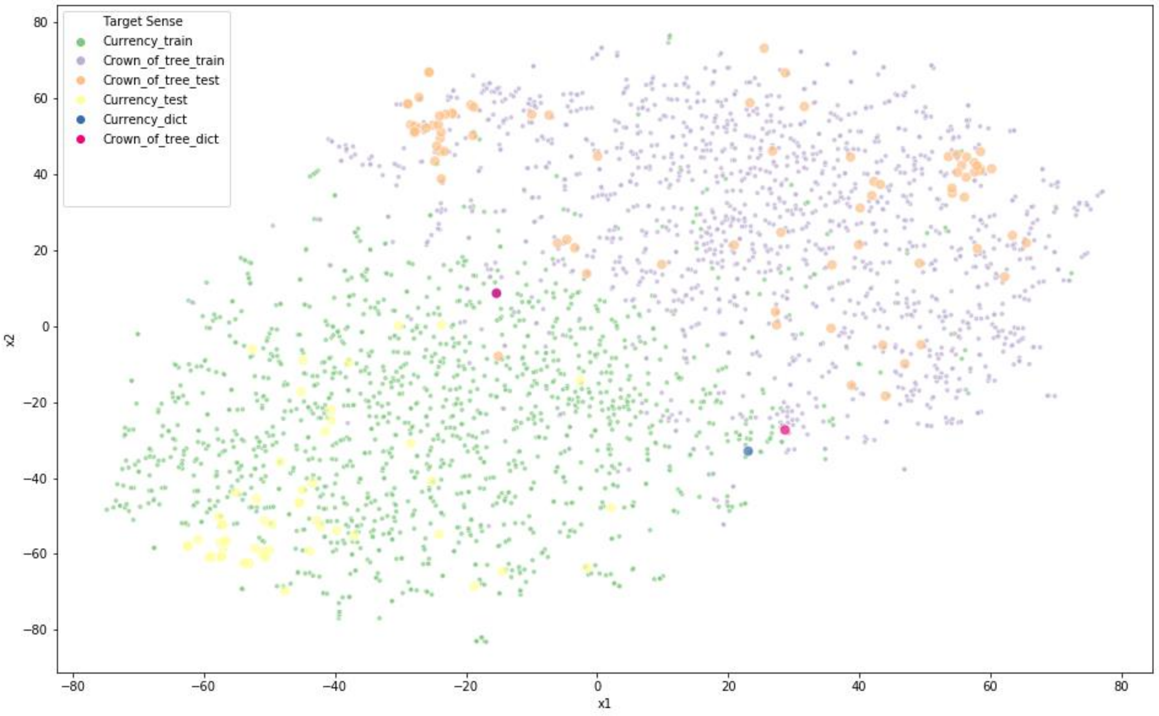

Figure 2: Representations for the word крона encoded by RusVectōrēs ELMo model, samples marked with “_train” label are taken from the Proza.ru train collection (balanced), examples marked with “_test” are taken from the manually annotated evaluation collection contexts, label “_dict” stands for the examples of use or dictionary definitions

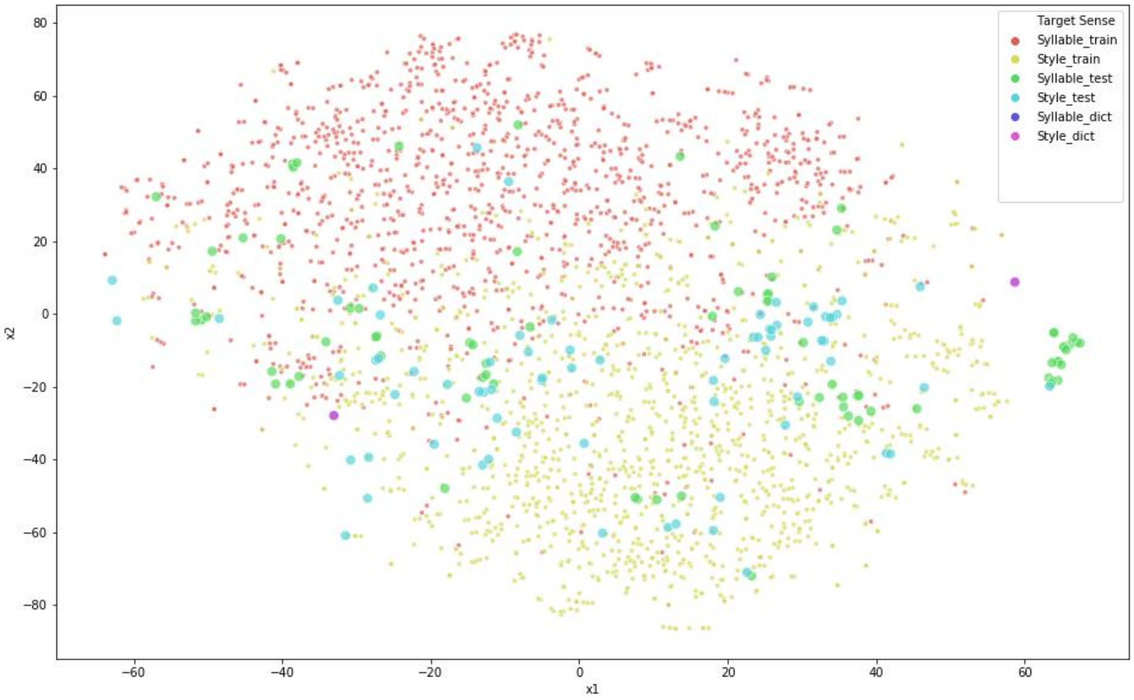

Figure 3: Representations for the word *слог* encoded by RusVectōrēs ELMo model, samples marked with "_train" label are taken from the Proza.ru train collection (balanced), examples marked with "_test" are taken from the manually annotated evaluation collection contexts, label "_dict" stands for the examples of use or dictionary definitions

7. Conclusion

In this article we introduced the method of automatic harvesting and labelling of the training collections that is aimed at mitigating knowledge acquisition bottleneck. This approach relies on the relations that connect target polysemous words and the monosemous words surrounding them in the semantic graph RuWordNet. In our algorithm the distances between the words under consideration in the thesaurus are not limited to the closest ones, which makes it possible for our algorithm to collect training samples for the vast majority of the polysemous words in the thesaurus. The procedure of the monosemous candidates ranking enables to add to the training collections only reliable samples thus reducing the amount of noise added to the training data.

The training collections were compiled from the texts extracted from the news and Proza.ru corpora. We evaluated them using kNN classifier applied to the contextualized word embeddings extracted for target polysemous words and measured its performance on the RUSSE-RuWordNet test dataset. The experiments showed the limitations and the benefits of different deep contextualized word representations to model polysemy. The best result on the generated text collections was obtained with the Proza.ru training collection and RusVectōrēs ELMo model and amounted to 0.857 F1 score.

Our experiments with the augmentation of the training collections demonstrated that the integration of any amount of hand-labelled data to the generated collection is beneficial for supervised models. The best result on the hybrid training data, which contains manually labelled and automatically generated samples, equals to 0.94. Furthermore, our research proved that lemmatized training data improves the performance of the WSD models for the languages with rich morphology such as Russian.

8. Acknowledgements

The work of Loukachevitch N. in the current study concerns formulation of the disambiguation approach for RuWordNet data, calculation of paths between synsets, criteria for selecting contexts. The study is supported by the Russian Science Foundation (project № 20-11-20166).

9. References

[1] G. A. Miller, C. Leacock, R. Tengi, R. T. Bunker, A semantic concordance, in: Proceedings of the workshop on Human Language Technology, pp. 303-308, 1993, Association for Computational Linguistics.

[2] K. Taghipour, H. T. N, One million sense-tagged instances for word sense disambiguation and induction, in: Proceedings of the nineteenth conference on computational natural language learning, pp. 338-344, 2015.

[3] D. Martinez, E. Agirre, X. Wang, Word relatives in context for word sense disambiguation, in: Proceedings of the Australasian Language Technology Workshop 2006, pp. 42-50, 2006.

[4] M. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, L. Zettlemoyer, Deep contextualized word representations, in: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 2227–2237, 2018.

[5] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 4171–4186, 2019.

[6] C. Leacock, G. A. Miller, M. Chodorow, Using corpus statistics and WordNet relations for sense identification. Computational Linguistics, vol. 24(1), pp. 147-165, 1998.

[7] G. Miller, WordNet: A Lexical Database for English. In: Communications of the ACM, vol.38(11), pp. 39-41, 1995.

[8] P. Przybyła, How big is big enough? Unsupervised word sense disambiguation using a very large corpus. arXiv preprint arXiv:1710.07960, 2017.

[9] R. Mihalcea, D. I. Moldovan, An Iterative Approach to Word Sense Disambiguation, in: FLAIRS Conference, pp. 219-223, 2000.

[10] H. C. Seo, H. Chung, H. C. Rim, S. H. Myaeng, S. H. Kim, Unsupervised word sense disambiguation using WordNet relatives, Computer Speech & Language, vol. 18, no. 3, pp. 253-273. SPEC. ISS, 2004.

[11] D. Yuret, KU: Word sense disambiguation by substitution, in: Proceedings of the 4th International Workshop on Semantic Evaluations, pp. 207-213, Association for Computational Linguistics, 2007.

[12] R. Mihalcea, Bootstrapping Large Sense Tagged Corpora, in: Proceedings of the Third International Conference on Language Resources and Evaluation (LREC-2002), vol. 1999, Las Palmas, Canary Islands, Spain, 2002.

[13] N. Loukachevitch, I. Chetviorkin, Determining the most frequent senses using Russian linguistic ontology RuThes, in: Proceedings of the workshop on Semantic resources and semantic annotation for Natural Language Processing and the Digital Humanities at NODALIDA 2015, pp. 21-27, 2015.

[14] V. Henrich, E. Hinrichs, T. Vodolazova, Webcage: a web-harvested corpus annotated with GermaNet senses, in: Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics, pp. 387-396. Association for Computational Linguistics, 2012.

[15] B. Scarlini, T. Pasini, R. Navigli, Just “OneSeC” for producing multilingual sense-annotated data, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pp. 699-709, 2019.

[16] E. Agirre, O. L. De Lacalle, Publicly Available Topic Signatures for all WordNet Nominal Senses, in: LREC, 2004.

[17] T. Pasini, R. Navigli, Train-o-Matic: Large-scale supervised word sense disambiguation in multiple languages without manual training data, in: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pp. 78-88, 2017.

[18] E. Barba, L. Procopio, N. Campolungo, T. Pasini, R. Navigli, MuLaN: Multilingual Label propagatioN for Word Sense Disambiguation, in: Proceedings of IJCAI, 2020.

[19] T. Pasini, The knowledge acquisition bottleneck problem in multilingual word sense disambiguation, in: Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-20, Yokohama, Japan, 2020.

[20] R. Navigli, Word sense disambiguation: A survey. ACM computing surveys (CSUR), vol. 41(2), 10, 2009.

[21] O. Melamud, J. Goldberger, I. Dagan, Context2vec: Learning Generic Context Embedding with Bidirectional LSTM, in: Proceedings. of COLING, pp. 51–61, 2016.

[22] A. Kutuzov, E. Kuzmenko, To lemmatize or not to lemmatize: how word normalisation affects ELMo performance in word sense disambiguation, in: Proceedings of the First NLPL Workshop on Deep Learning for Natural Language Processing, pp. 22-28, 2019.

[23] G. Wiedemann, S. Remus, A. Chawla, C. Biemann, Does BERT Make Any Sense? Interpretable Word Sense Disambiguation with Contextualized Embeddings, arXiv preprint arXiv:1909.10430, 2019.

[24] J. Du, F. Qi, M. Sun, Using BERT for word sense disambiguation, in: arXiv preprint arXiv:1909.08358, 2019.

[25] C. Hadiwinoto, H. T. Ng, W. C. Gan, Improved word sense disambiguation using pre-trained contextualized word representations, in: arXiv preprint arXiv:1910.00194, 2019.

[26] N. V. Loukachevitch, G. Lashevich, A. A. Gerasimova, V. V. Ivanov, B. V. Dobrov, Creating Russian WordNet by Conversion, in: Proceedings of Conference on Computational linguistics and Intellectual technologies Dialog-2016, pp. 405-415, 2016.

[27] T. Shavrina, O. Shapovalova, To the methodology of corpus construction for machine learning: «Taiga» syntax tree corpus and parser, in: Proceedings of "CORPORA2017", international conference, Saint-Petersbourg, 2017.

[28] A. Panchenko, A. Lopukhina, D. Ustalov, K. Lopukhin, N. Arefyev, A. Leontyev, N. Loukachevitch, RUSSE'2018: A Shared Task on Word Sense Induction for the Russian Language, in: Computational Linguistics and Intellectual Technologies: Papers from the Annual International Conference "Dialogue", Moscow, Russia. RSUH, pp. 547–564, 2018.

[29] A. A. Lopukhina, V. Yu. Apresyan, B. L. Iomdin, Yu. D. Apresyan, O. Yu. Boguslavsaya, T. V. Krylova, I. B. Levontina, A. V. Sannikov, E. V. Uryson, E. E. Babaeva, M. Ya. Glovinskaya, A. V. Ptentsova, Active Dictionary of the Russian Language [Aktivnyj slovar' russkogo yazyka]. Publishing House Nestor-Istoria, Moscow, Vol. 3, 2017.

[30] S.I. Ozhegov, Explanatory Dictionary of the Russian Language. Ed. by Skvortsova S.I., 8, pp. 1376, 2014.

[31] A. Bolshina, N. Loukachevitch, Generating training data for word sense disambiguation in Russian, in: Proceedings of Conference on Computational linguistics and Intellectual technologies Dialog-2020, pp. 119-132, 2020.

[32] T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient Estimation of Word Representations in Vector Space, in Proceedings of Workshop at ICLR, 2013.

[33] M. Korobov, Morphological Analyzer and Generator for Russian and Ukrainian Languages, in: Analysis of Images, Social Networks and Texts, pp. 320-332, 2015.

[34] A. Kutuzov, E. Kuzmenko, WebVectors: A Toolkit for Building Web Interfaces for Vector Semantic Models, in: Ignatov D. et al. (eds) Analysis of Images, Social Networks and Texts. AIST 2016. Communications in Computer and Information Science, vol 661. Springer, Cham, 2017.

[35] Y. Kuratov, M. Arkhipov, Adaptation of Deep Bidirectional Multilingual Transformers for Russian Language, arXiv preprint arXiv:1905.07213, 2019.

[36] M. Lesk, Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone, in: Proceedings of the International Conference on Systems Documentation, 1986.

[37] F. Luo, T. Liu, Q. Xia, B. Chang, Z. Sui, Incorporating glosses into neural word sense disambiguation, arXiv preprint arXiv:1805.08028, 2018.

[38] L. Huang, C. Sun, X. Qiu, X. Huang, GlossBERT: BERT for word sense disambiguation with gloss knowledge, arXiv preprint arXiv:1908.07245, 2019.

[39] T. Blevins, L. Zettlemoyer, Moving Down the Long Tail of Word Sense Disambiguation with Gloss-Informed Biencoders, arXiv preprint arXiv:2005.02590, 2020.

[40] D. Loureiro, A. Jorge, Language modelling makes sense: Propagating representations through wordnet for full-coverage word sense disambiguation, in: arXiv preprint arXiv:1906.10007, 2019.

[41] A. Bolshina, N. Loukachevitch, Automatic Labelling of Genre-Specific Collections for Word Sense Disambiguation in Russian, Russian Conference on Artificial Intelligence, Springer, Cham, pp. 215-227, 2020.