Computational Linguistics and Intellectual Technologies:
Proceedings of the International Conference "Dialogue 2020"

Moscow, June 17–20, 2020

GENERATING TRAINING DATA FOR WORD SENSE DISAMBIGUATION IN RUSSIAN

Bolshina A. S. (angelina_ku@mail.ru)
Lomonosov Moscow State University, Moscow, Russia

Loukachevitch N. V. (louk_nat@mail.ru)
Kazan Federal University, Kazan, Russia

The best approaches in Word Sense Disambiguation (WSD) are supervised and rely on large amounts of hand-labelled data, which is not always available and costly to create. For the Russian language there is no sense-tagged resource of the size sufficient to train supervised word sense disambiguation algorithms. In our work we describe an approach that is used to create an automatically labelled collection based on the monosemous relatives (related unambiguous entries). The main contribution of our work is that we extracted monosemous relatives that can be located at relatively long distances from a target ambiguous word and ranked them according to the similarity measure to the target sense. The selected candidates are then used to extract training samples from the news corpus. We evaluated word sense disambiguation models based on a nearest neighbor classification on BERT and ELMo embeddings. Our work relies on the Russian wordnet RuWordNet.

Keywords: Word sense disambiguation, Russian dataset, Monosemous relatives, Automatic Dataset Collection, ELMo, BERT

DOI: 10.28995/2075-7182-2020-19-119-132

АВТОМАТИЧЕСКИЙ СБОР И РАЗМЕТКА ОБУЧАЮЩЕЙ КОЛЛЕКЦИИ ДЛЯ ЗАДАЧИ РАЗРЕШЕНИЯ ЛЕКСИЧЕСКОЙ НЕОДНОЗНАЧНОСТИ НА РУССКОМ ЯЗЫКЕ

Большина А. С. (angelina_ku@mail.ru)
МГУ им. М. В. Ломоносова, Москва, Россия

Лукашевич Н. В. (louk_nat@mail.ru)
Казанский федеральный университет, Казань, Россия

Ключевые слова: разрешение лексической неоднозначности, обучающая коллекция текстов на русском языке, однозначные родственники, автоматический сбор текстовой коллекции, ELMo, BERT

1. Introduction

Word Sense Disambiguation (WSD) is a key task in semantic-oriented applications such as semantic text analysis, semantic information retrieval and knowledge graph construction. To achieve high performance, supervised WSD algorithms require large sense-annotated datasets. The annotation of such corpora demands considerable time and human resources, that is why supervised machine learning approaches suffer from a knowledge acquisition bottleneck.

There exist several hand-crafted sense-annotated datasets for English ([Miller et al. 1993], [Taghipour & Ng 2015]). However, not for all languages such corpora are available, and that certainly hinders the development of WSD systems for those languages. This also holds true for the Russian language.

Automatic acquisition of training samples can help to tackle this problem. Our research is focused on the monosemous relatives approach, which exploits a set of unambiguous words (or phrases) related to particular senses of polysemous word. However, as it was noted in [Martinez et al. 2006], 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 and used word embeddings to estimate the similarity between a monosemous relative and a particular sense of a target word, which is further used in the development of the training collection. According to our knowledge, this is the first work that uses embedding similarity for selection of training contexts for the word sense disambiguation task. In order to evaluate the created training collections, we utilized contextualized word representations—ELMo [Peters et al. 2018] and BERT [Devlin et al. 2019]. We also conducted an experiment to compare the performance of the models trained on the collections with close monosemous relatives (synonyms, hypernyms and hyponyms) and more distant ones[1].

The paper is organized as follows. In section two we review the related work. Section three describes the data utilized in the research. The fourth section describes the method applied to automatically generate and annotate training collections. The procedure of creating a collection is explained in the fifth section. In the sixth section we describe a supervised word sense disambiguation algorithm trained on our collected material and also present the results obtained by four different models. 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 [Leacock et al. 1998]. Usually WordNet [Miller 1995] is used as a source for such relatives. WordNet is a lexical-semantic resource for the English language that contains description of nouns, verbs, adjectives, and adverbs in 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 [Przybyła 2017]. In the work [Martinez et al. 2006] 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.

Some researchers replace the target word with named entities [Mihalcea & Moldovan 2000], some researchers substitute it with meronyms and holonyms [Seo et al. 2004]. In the article [Yuret 2007] 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 [Mihalcea 2002], that is used to construct 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 [Loukachevitch & Chetviorkin 2015].

Other methods of automatic generation of training collections for WSD exploit parallel corpora [Taghipour & Ng 2015], Wikipedia and Wiktionary [Henrich et al. 2012], topic signatures [Agirre & De Lacalle 2004]. [Pasini & Navigli 2017] created large training corpora exploiting a graph-based method that took an unannotated corpus and a semantic network as an input.

[1] The source code of our algorithm and experiments is publicly available at: https://github.com/loenmac/russian_wsd_data.

Various supervised methods including kNN, Naive Bayes, SVM, neural networks are applied to word sense disambiguation [Navigli 2009]. Recent studies have shown the effectiveness of contextualized word representations for the WSD task ([Wiedemann et al. 2019], [Kutuzov & Kuzmenko 2019]). The most widely used deep contextualized embeddings are ELMo [Peters et al. 2018] and BERT [Devlin et al. 2019]. In ELMo (Embeddings from language models) [Peters et al. 2018] 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) [Devlin et al. 2019] has a different type of architecture, namely multi-layer bidirectional Transformer encoder. During pre-training procedure, the model is "jointly conditioning on both left and right context in all layers" [Devlin et al. 2019]. 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, fit well into the task of WSD, we employ them in our investigation.

3. Data

In our research as an underlying semantic network we exploit Russian wordnet RuWordNet [Loukachevitch et al. 2016]. It is a semantic network for Russian that has a WordNet-like structure. It is composed of 111.5 thousand of words and word combinations for the Russian language. RuWordNet has been published on the Linguistic Linked Open Data cloud [Cimiano et al. 2020] and interlinked [Loukachevitch & Gerasimova 2019] with the Collaborative Interlingual Index (CILI) [Bond et al. 2016].

RuWordNet contains 29,297 synsets for nouns. There are 63,014 monosemous and 5,892 polysemous nouns in RuWordNet. Total number of polysemous nouns' senses equals to 14,357. This resource was used to extract semantic relations (e.g. synonymy, hyponymy etc.) between a target sense of a polysemous word and all the words (phrases) connected to it, including those linked via distant paths. The sense inventory was also taken from this resource.

As a reference corpus we utilized a news corpus, that consists of one million news articles harvested from various news sources. All the texts have already been cleaned from html-elements or any other markup. The corpus consists of 24.2 million sentences, 288,1 million lemmas and 1,4 million of unique lemmas.

For evaluation of our algorithm of training data generation, we used three distinct RUSSE'18 datasets for Russian [Panchenko et al. 2018]. 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 [Apresyan et al. 2017] 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 words whose senses are absent in RuWordNet. For example, the word *гипербола* 'hyperbole' from RUSSE'18 dataset is missing in the

RuWordNet vocabulary. The word *мандарин* has two senses described in RUSSE'18: its sense 'tangerine' is included in the thesaurus, whereas its sense 'mandarin, bureaucrat' is absent, that is why we did not put it in the final test set. Some of the words like *карьер* 'quarry/a very fast gallop' and *шах* 'shah/check' do not have enough examples for one of their senses in the news corpus.

The final list of the target ambiguous words contains 30 words in total, each having two different senses. The **Appendix 1** contains the list of the target ambiguous words selected from RUSSE'18 dataset. For convenience we will call the resulting test dataset RUSSE-RuWordNet because it is a projection of RUSSE'18 sense inventory on the RuWordNet data. The entire dataset consists of 2,103 sentences, 39,311 lemmas and 12,110 unique lemmas.

We also created a small training dataset, that consists of the word sense definitions and examples of uses from Ozhegov dictionary [Ozhegov 2014] for every target polysemous word. Each sense of an ambiguous word has one definition and between 1 and 3 usage examples. This training data is utilized as a baseline for the WSD task.

4. Candidate Selection and Ranking Algorithm

The central idea of our method is based on the assumption that a training collection can be built not only with the direct relations like synonymy, hyperonymy and hyponymy but also with far more distant words, such as co-hyponyms. For example, most contexts for the word *крона* in the sense 'krona, currency' match the contexts of the other words denoting currency like *английский фунт* 'pound sterling' as they have common hypernym *валюта* 'currency'.

The principal features of our approach are as follows:

1. We take into consideration not only the closest relatives to a target word sense, as it was done in previous works, but also more distant relatives.
2. We utilize similarity scores between a candidate monosemous relative and synsets close to a sense of a target polysemous word in order to evaluate how well this candidate can represent the sense of an ambiguous word.
3. We introduce the notion of ***a synset nest*** which is used to assess the potential of candidate's usage contexts for displaying target sense of a polysemous word. To measure the relevance and suitability of a monosemous candidate, we exploit a thesaurus set of words similar to a target sense. The group of synonyms to a target sense and all the words from directly related synsets within 2 steps from a target word comprise ***the synset nest*** for a target sense.
4. We check similarity scores to the synset nest for both closest and further located monosemous relatives because a word described as monosemous in the thesaurus can actually have polysemous usage in a corpus. For example, Russian word *ириска* ('toffee') can also denote a nickname of Everton Football Club (The Toffees) [Loukachevitch 2019]. Thus, all candidate monosemous relatives should be further checked on the source corpus.
5. We propose two distinct methods of compiling a training collection based on the monosemous relatives rating.

A target word sense is a sense of a polysemous word that we want to disambiguate. Candidate monosemous relatives are unambiguous words and phrases, that can be located in up to four-step relation paths to a polysemous word. Candidate monosemous relatives are unambiguous words and phrases which can be located in up to four-step relation paths to a polysemous word. We consider only those words or word combinations, that have more than 50 occurrences in the news corpus.

A fragment of the synset 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…'

Our method of extracting monosemous relatives is based on comparison of distributional and thesaurus similarities. The word embedding model is utilized to select the most appropriate monosemous relatives whose context serve as a good representation of a target word sense. We used the word2vec model to extract 100 most similar words to each monosemous word from the candidates list. Thus, we collected the words that represent a distributional set of close words with the respective cosine similarities measures. 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 s_j.
2. We compile the synset nest ns_j which consists of all closely related words to the target sense s_j, that is, for example, synonyms, hyponyms, hypernyms and cohyponyms. The synset nest ns_j consists of N_k synsets.
3. For each candidate monosemous relative r_j, we find 100 most similar words according to the word2vec model trained on a reference corpus.
4. We intersect these top-100 words with the words included in the synset nest ns_j of the target sense 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 synset nest s_j is determined by the maximum weight among the words $w_{k_1}^j, \dots, w_{k_i}^j$ representing this synset in the intersection.
6. The total score of the monosemous candidate r_j is the sum of the weights of all synsets from the synset nest ns_j. In such a way more scores are assigned to those candidates, that resemble a greater number of synsets from the synset nest close the target sense of the ambiguous target word. Thus, the final weight of the candidate can be defined as follows:

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

The following fragment of list of monosemous relatives with similarity scores (given in brackets) was obtained for the noun *гвоздика* 'clove':

(2) *чёрный перец* 'black pepper' (7.5), *кардамон* 'cardamom' (6.8), *корица* 'cinnamon' (6.5), *имбирь* 'ginger' (6.4), *мускатный орех* 'nutmeg' (6) ...

We have also found some examples where a monosemous word is connected to a sense of a target word but recieved zero similarity weight. For example, the word *марля* 'gauze' is a cohyponym to the word *байка* in the sense 'thick flannelette', but it was not included in the monosemous relatives list because its distributional set of close words did not have any intersection with the synset nest.

As a result of the described procedure, all monosemous relatives are sorted by the weight they obtained. The higher rated monosemous relatives are supposed to be better candidates to represent the sense of the target word and, consequently, their contexts of use are best suited as the training examples in the WSD task. The candidate ranking algorithm identifies which monosemous relatives are most similar to the target ambiguous word's sense. Once we have selected the monosemous candidates, we can extract from the corpus the contexts in which they occur. Then we substitute the monosemous relatives with the target ambiguous word in these texts and add them to a training collection.

5. Generating Training Data using Monosemous Relatives

The news corpus was used to extract the contexts with monosemous relatives found by the proposed algorithm. For comparison, we decided to create training collections in two ways. We compiled the first collection 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.

As for the second collection, 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. Accordingly, 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.

The quantitative characteristics of the relations connecting the target senses and their monosemous relatives, distances between them and a proportion of monosemous relatives expressed as a phrase are given in the **Table 1**.

Table 1. Quantitative characteristics of monosemous relatives

Feature	Proportion of occurrences
Distance to a target sense	
0 (synset)	2%
1	13%
2	38%
3	31%
4	16%

Feature	Proportion of occurrences
Relation between a target sense and a monosemous relative	
Synonyms	2%
Hyponyms	13%
Hypernyms	11%
Cohyponyms	28%
Cohyponyms situated at three-step path	24%
Cohyponyms situated at four-step path	19%
Other	3%
Word combinations	48%

The word2vec embedding model that we used in our experiments was trained on the news corpus with the window size of 3. As a preprocessing step, we split the corpus into separate sentences, tokenized them, removed all the stop words and lemmatized the words with pymorphy2 tool [Korobov 2015]. We decided to lemmatize the train and test data, because it was shown in [Kutuzov & Kuzmenko 2019, 2], that "feeding ELMo with lemmas instead of raw tokens can improve WSD performance". The words obtained from the word2vec model were filtered out—we removed the ones not included in the thesaurus vocabulary.

In the Appendix 1 we present the characteristics of the two training collections: the list of the target ambiguous words selected from RUSSE'18 dataset, their senses and the number of examples per each sense respective to the collection type.

6. Experiments

We conducted several experiments to determine whether our text collection can be used as a training dataset for a WSD model. Following [Wiedemann et al. 2019], 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.

In our experiments we exploited two distinct ELMo models—the one trained by DeepPavlov on Russian WMT News and the other is RusVectōrēs [Kutuzov & Kuzmenko 2017] lemmatized ELMo model trained on Taiga Corpus [Shavrina & Shapovalova 2017]. The difference between these two models is that from the first model we extracted a vector for a whole sentence with a target word, whereas from the second model we extracted a single vector for a target ambiguous word. As for BERT, we used two 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 [Kuratov & Arkhipov 2019]. To extract BERT contextual representations, we followed the method described by [Devlin et al. 2019] and [Wiedemann et al. 2019] and concatenated "the token representations from the top four hidden layers of the pre-trained Transformer" [Devlin et al. 2019].

The **Table 2** demonstrates the results obtained by different types of contextualized word embeddings, the training collections and model parameters. As it can clearly be seen, all the systems surpassed the quality level of the baseline solution trained on the dataset of the dictionary definitions and usage examples. So, this means that we have managed not only to collect training data sufficient to train the WSD model but also to show a good performance on the RUSSE-RuWordNet dataset.

Table 2. F1 scores for ELMo- and BERT-based WSD models (best results are marked bold): (k)—number of nearest neighbors, (1)—Corpus-1000, (2)—Balanced collection

Model	ELMo RusVectōrēs (target word)		ELMo DeepPavlov (whole sentence)		RuBERT DeepPavlov		Multilingual BERT	
(k)	(1)	(2)	(1)	(2)	(1)	(2)	(1)	(2)
1	0.794	0.797	**0.752**	0.758	0.735	0.75	0.67	0.662
3	0.811	0.81	0.749	0.753	0.756	0.755	0.673	0.681
5	**0.819**	0.81	0.748	0.756	0.771	0.769	0.667	0.682
7	**0.819**	0.815	0.746	**0.759**	**0.774**	0.768	0.673	0.683
9	0.816	**0.821**	0.747	0.753	0.769	**0.774**	**0.677**	**0.688**
Baseline	0.772		0.716		0.667		0.672	

The qualitive analysis of the classification errors showed that the main cause of mistakes were lexical and structural differences between the training and test sets. The examples from the test dataset were taken from the Russian National Corpus and Wikipedia, whereas the training collections were composed of news articles. Adding more data of various genres will help to diversify the training collections, thus, the training samples will have more similar representations to the test ones.

The algorithm based on ELMo pre-trained embeddings by RusVectōrēs outperformed all other models achieving 0.821 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 minor performance drop for the Corpus-1000 for all the models except for the RuBERT model. Corpus-1000 does not include all possible monosemous relatives, so dataset lacks contextual diversity, the balanced collection, on the contrary, is more representative with regard to the variety of contexts.

It is worth noting, that only 20 words of our dataset have close-related monosemous relatives for all their senses connected to a target word with the direct relations (synonymy, hyponymy, hypernymy), which means that the proposed expansion of paths is very useful. We evaluated the results of word sense disambiguation for these 20 words using only direct monosemous relatives and all the proposed relatives and found that the best results achieved by the RusVectōrēs ELMO model are quite similar: 0.841 (direct relatives) vs. 0.835 (all relatives).

7. Conclusion

The issue that we addressed in this article is the lack of sense-annotated training data for supervised WSD systems in Russian. In this paper we have described our algorithm of automatic collection and annotation of training data for the Russian language. Our training collections consist of the texts obtained from the news corpus and can be further replenished. The main contribution of the paper is that we have considered in the selection algorithm a wide range of monosemous relatives' types and utilized the metric based on a cosine similarity to determine the most appropriate monosemous relatives to be added to the training collection.

In order to evaluate the training collections, we applied kNN classifier to the contextualized word embeddings extracted for the target polysemous words and measured its performance on the RUSSE-RuWordNet test dataset. We have investigated the capability of different deep contextualized word representations to model polysemy. The best result was obtained with RusVectōrēs ELMo model and amounted to 0.821 F1 score.

As future work we plan to add more texts of different genres to the training collection.

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; this work is supported by the Russian Science Foundation grant no. 19-71-10056 financed through Kazan Federal University.

References

1. *Agirre E., De Lacalle O. L.* (2004), Publicly Available Topic Signatures for all WordNet Nominal Senses. In LREC.
2. *Bond F., Vossen P., McCrae J.P., Fellbaum C.* (2016), CILI: the Collaborative Interlingual Index. In Proceedings of the 8th Global WordNet Conference (GWC 2016), pp. 50–57.
3. *Cimiano P., Chiarcos C., McCrae J.P., Gracia J.* (2020), Linguistic Linked Open Data Cloud. In Cimiano P., et al. Linguistic Linked Data: Representation, Generation and Applications. Springer, pp. 29–41.
4. *Devlin J., Chang M.-W., Lee K., Toutanova K.* (2019), 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.
5. *Henrich V., Hinrichs E., Vodolazova T.* (2012), 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.

6. *Korobov M.* (2015), Morphological Analyzer and Generator for Russian and Ukrainian Languages. In: Analysis of Images, Social Networks and Texts, pp. 320–332.
7. *Kuratov Y., Arkhipov M.* (2019), Adaptation of Deep Bidirectional Multilingual Transformers for Russian Language. arXiv preprint arXiv:1905.07213.
8. *Kutuzov A., Kuzmenko E.* (2017), 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.
9. *Kutuzov A., Kuzmenko E.* (2019), 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.
10. *Leacock C., Miller G. A., Chodorow M.* (1998), Using Corpus Statistics and WordNet Relations for Sense Identification. Computational Linguistics, vol. 24(1), pp. 147–165.
11. *Lopukhina A. A., Apresyan V. Yu., Iomdin B. L., Apresyan Yu. D., Boguslavsaya O. Yu., Krylova T. V., Levontina I. B., Sannikov A. V., Uryson E. V., Babaeva E. E., Glovinskaya M. Ya., Ptentsova A. V.* (2017), Active Dictionary of the Russian Language [Aktivnyj slovar' russkogo yazyka]. Publishing House Nestor-Istoria, Moscow, Vol. 3.
12. *Loukachevitch N. V.* (2019), Corpus-based Check-up for Thesaurus. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pp. 5773–5779.
13. *Loukachevitch N. V., Lashevich G., Gerasimova A. A., Ivanov V. V., Dobrov B. V.* (2016), Creating Russian WordNet by Conversion. In Proceedings of Conference on Computational linguistics and Intellectual technologies Dialog-2016, pp. 405–415.
14. *Loukachevitch N. V., Chetviorkin I.* (2015), 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.
15. *Loukachevitch N., Gerasimova A.* (2019), Linking Russian Wordnet RuWordNet to WordNet. In Proceedings of the 10th Global Wordnet Conference (GWC 2019), pp. 64–71.
16. *Martinez D., Agirre E., Wang X.* (2006), Word Relatives in Context for Word Sense Disambiguation. In Proceedings of the Australasian Language Technology Workshop 2006, pp. 42–50.
17. *Mihalcea R.* (2002), 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.
18. *Mihalcea R., Moldovan D. I.* (2000), An Iterative Approach to Word Sense Disambiguation. In FLAIRS Conference, pp. 219–223.
19. *Miller G.* (1995), WordNet: A Lexical Database for English. In Communications of the ACM, vol. 38(11), pp. 39–41.

20. *Miller G. A., Leacock C., Tengi R., Bunker R. T.* (1993), A Semantic Concordance. In Proceedings of the workshop on Human Language Technology, pp. 303–308. Association for Computational Linguistics.
21. *Navigli R.* (2009), Word Sense Disambiguation: A survey. ACM computing surveys (CSUR), vol. 41(2), 10.
22. *Ozhegov S. I.* (2014), Explanatory Dictionary of the Russian Language [Tolkovy Slovar' Russkogo Yazyka]. Edited by Skvortsova S. I., 8, pp. 1376.
23. *Panchenko A., Lopukhina A., Ustalov D., Lopukhin K., Arefyev N., Leontyev A., Loukachevitch N.* (2018), 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.
24. *Pasini T., Navigli R.* (2017), 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.
25. *Peters M., Neumann M., Iyyer M., Gardner M., Clark C., Lee K., Zettlemoyer L.* (2018), 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.
26. *Przybyła P.* (2017), How Big is Big Enough? Unsupervised word sense disambiguation using a very large corpus. arXiv preprint arXiv:1710.07960.
27. *Seo H. C., Chung H., Rim H. C., Myaeng S. H., Kim S. H.* (2004), Unsupervised Word sense Disambiguation Using WordNet Relatives. Computer Speech & Language SPEC. ISS., vol. 18, no. 3, pp. 253–273.
28. *Shavrina T., Shapovalova O.* (2017), To the Methodology of Corpus Construction for Machine Learning: «Taiga» Syntax Tree Corpus and Parser. In Proceedings of "CORPORA2017", international conference, Saint-Petersburg.
29. *Taghipour K., Ng H. T.* (2015), 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.
30. *Wiedemann G., Remus, S., Chawla A., Biemann C.* (2019), Does BERT Make Any Sense? Interpretable Word Sense Disambiguation with Contextualized Embeddings. arXiv preprint arXiv:1909.10430.
31. *Yuret D.* (2007), KU: Word Sense Disambiguation by Substitution. In Proceedings of the 4th International Workshop on Semantic Evaluations, pp. 207–213. Association for Computational Linguistics.

Приложение 1. Appendix

Appendix 1. Target polysemous words from RUSSE-RuWordNet dataset

Polysemous word	Sense	Corpus-1000	Balanced collection
акция$_1$	Share/stock	1000	1239
акция$_2$	Action	1000	1314
байка$_1$	Tale/story	1000	1227
байка$_2$	Thick flannelette	245	245
гвоздика$_1$	Carnation	1000	1314
гвоздика$_2$	Cloves	1000	1154
гусеница$_1$	Caterpillar	1000	1295
гусеница$_2$	Track	1000	1153
капот$_1$	Bonnet/hood of a car	1000	918
капот$_2$	Housecoat	1000	1084
крона$_1$	Top of a tree	1000	1131
крона$_2$	Krona (currency)	1000	1314
рок$_1$	Rock music	1000	1016
рок$_2$	Destiny	1000	938
слог$_1$	Syllable	1000	1047
слог$_2$	Style	1000	1137
стопка$_1$	Pile	1000	1258
стопка$_2$	Small drinking glass	1000	1005
таз$_1$	Pelvis	1000	1124
таз$_2$	Basin	1000	1314
такса$_1$	Price/charge	1000	1300
такса$_2$	Dachshund	1000	1069
замок$_1$	Castle	1000	1078
замок$_2$	Lock	1000	947
лук$_1$	Bow	1000	1286
лук$_2$	Onion	1000	1267
бор$_1$	Boron	1000	1292
бор$_2$	Pine Forest	1000	675
дар$_1$	Talent	1000	1117
дар$_2$	Gift	1000	1169
двигатель$_1$	Engine	1000	1310
двигатель$_2$	Something that causes a process to happen	1000	1305
дедушка$_1$	Old man	1000	1299
дедушка$_2$	Grandfather	1000	1231
декрет$_1$	Maternity leave	128	128

Polysemous word	Sense	Corpus-1000	Balanced collection
декрет$_2$	Decree	1000	1300
дерево$_1$	Tree	1000	1309
дерево$_2$	Timber	1000	966
диалог$_1$	Conversation	1000	1278
диалог$_2$	Negotiations	1000	1300
диплом$_1$	Certificate	1000	1253
диплом$_2$	Diploma paper	1000	1246
доктор$_1$	Doctor, physician	1000	1310
доктор$_2$	Doctor, degree	1000	1300
доля$_1$	Part	1000	1300
доля$_2$	Destiny	1000	1300
достижение$_1$	Achievement	1000	1300
достижение$_2$	Reaching the level	1000	1309
жестокость$_1$	Ruthlessness	1000	801
жестокость$_2$	Cruelty	1000	1313
жребий$_1$	Lot	1000	1280
жребий$_2$	Destiny	1000	1300
затея$_1$	Fun	1000	1308
затея$_2$	Enterprise	1000	1309
застой$_1$	Stasis	1000	758
застой$_2$	Stagnation	1000	1235
затишье$_1$	Decline in activity	1000	1206
затишье$_2$	Calm	1000	1300
затмение$_1$	Mental breakdown	1000	1300
затмение$_2$	Eclipse	1000	1002