

Computational Linguistics and Intellectual Technologies:  
Proceedings of the International Conference “Dialogue 2016”

Moscow, June 1–4, 2016

## **РАЗРЕШЕНИЕ ЛЕКСИЧЕСКОЙ МНОГОЗНАЧНОСТИ ДЛЯ РУССКИХ ГЛАГОЛОВ С ИСПОЛЬЗОВАНИЕМ СЕМАНТИЧЕСКИХ ВЕКТОРОВ И СЛОВАРНЫХ ДАННЫХ**

**Лопухин К. А.** (kostia.lopuhin@gmail.com)

Scrapinghub, Москва, Россия

**Лопухина А. А.** (nastya-merk@yandex.ru)

Институт русского языка имени  
В. В. Виноградова РАН, Москва, Россия.

**Ключевые слова:** полисемия, многозначность, разрешение лексической многозначности, частоты значений, семантические вектора

## **WORD SENSE DISAMBIGUATION FOR RUSSIAN VERBS USING SEMANTIC VECTORS AND DICTIONARY ENTRIES**

**Lopukhin K. A.** (kostia.lopuhin@gmail.com)

Scrapinghub, Moscow, Russia

**Lopukhina A. A.** (nastya-merk@yandex.ru)

V. V. Vinogradov Russian Language Institute of the Russian  
Academy of Sciences, Moscow, Russia

Word sense disambiguation (WSD) methods are useful for many NLP tasks that require semantic interpretation of input. Furthermore, such methods can help estimate word sense frequencies in different corpora, which

is important for lexicographic studies and language learning resources. Although previous research on Russian polysemous verbs disambiguation established some important and interesting results, it was mostly focused on reducing ambiguity or determining the most frequent sense, but not on evaluating WSD accuracy. To the best of our knowledge, there is no comprehensively evaluated method that can perform semi-supervised word sense disambiguation for Russian verbs. In this paper we present a WSD method for verbs that is able to reach an average disambiguation accuracy of 75% using only available linguistic resources: examples and collocations from the Active Dictionary of Russian and large unlabeled corpora. We evaluate the method on contexts sampled from the web-based corpus RuTenTen11 for 10 verbs with 100 contexts for each verb. We compare different variations of the method and analyze its limitations. Method's implementation and labeled contexts are available online.

**Key words:** polysemy, word sense disambiguation, sense frequency, word2vec, semantic vectors

## 1. Introduction

Lexical-semantic ambiguity is an inherent property of any natural language, thus word sense disambiguation (WSD) is an important part of many natural language processing tasks. Various WSD techniques were discussed during SemEval sessions (Pradhan et al. 2007) and in WSD surveys (Ide and Véronis 1998; Navigli 2009; Mihalcea 2011). The most powerful and promising approaches are those that use already existing resources and do not require much human labeled data. Knowledge-based approaches take advantage of thesauri, for example English WordNet (Fellbaum 1998) or Russian Ru-Thes (Loukachevitch and Chujko 2007; Loukachevitch and Dobrov 2007) and encyclopedic resources, like Wikipedia (Ponzetto and Navigli 2010) and can be applied to domain-specific corpora with high accuracy (Agirre et al. 2009). Unsupervised corpus-based approaches typically perform clustering of senses in a corpus without making explicit references to any sense inventory, see e.g. (Schutze 1998; Huang 2012; Neelakantan 2014; Bartunov et al. 2015).

For Russian, several WSD experiments were performed on the Russian National corpus (RNC, ruscorpora.ru). Kobritsov et al. (2005) discussed the problem of automated word sense tagging in a large corpus and proposed a WSD approach based on lexical context markers. Shemanayeva et al. (2007) developed semantic filters aimed at raising the accuracy of sense disambiguation for adjectives in RNC. Mitrofanova et al. (2008) compared WSD techniques for Russian nouns that take into account either word lexical contexts or lexical-semantic tags of words in context. Both methods' average accuracies reached 83% and 85% respectively, which is comparable to the state-of-the-art in this field.

Sense disambiguation for Russian verbs was based on various linguistic resources. Kobritsov et al. (2007) performed a series of experiments on word sense disambiguation for verbs using information automatically extracted from dictionaries. They concluded that although the government pattern proves useful for disambiguation,

automatically extracted information does not provide any substantial reduction of polysemy, and that accounting for semantic properties of the arguments is the most promising approach. Similar results were discussed in (Kustova and Toldova 2008). They studied several different methods of decreasing polysemy for Russian verbs: government pattern (morphological properties of arguments) and semantic properties of the arguments. Government pattern helped to halve the number of possible senses, and using fine-grained semantic properties of arguments allowed to reduce the number of possible senses to a single one for most studied verbs and most contexts.

Theoretical studies of verb polysemy prove that valencies and government patterns normally stay the same as regular metaphoric shifts take place (Rozina 2005; Reznikova 2014). Valencies usually change if new meanings appear in slang, for example *X gonit Y iz Z* ‘X produces substance Y from substance Z’ / *X gonit* ‘X lies or says nonsense’ from the first issue of the Active Dictionary of Russian (Apresjan et al. 2014). For our WSD experiments, which imply disambiguation for all verb senses, it seems reasonable to focus on lexical contexts of verbs rather than on their syntactic structure.

Other projects related to WSD of verbs are Disambiguation of Verbs by Collocation and Corpus Pattern Analysis led by Patrick Hanks and colleagues. These projects are focused on statistical analysis of corpus data in order to discover typical usage patterns and create the Pattern Dictionary of English Verbs (<http://pdev.org.uk/>; Hanks and Pustejovsky 2005; Hanks 2008). The authors emphasize that meanings are associated with prototypical sentence contexts (patterns or collocations) and not with word senses from dictionaries. Cf. also (Gries et al. 2010), where frequency distributions of English verbal constructions are discussed. The ongoing project of Russian FrameBank also focuses on verb constructions (<http://framebank.ru/>; Lyashevskaya 2012; Kashkin and Lyashevskaya 2013). Although the abovementioned methods presuppose disambiguation techniques and are corpus-based, they deal with collocations and not dictionary senses and we do not consider them in our study.

In this paper we set out to evaluate several techniques of word sense disambiguation for Russian verbs. The techniques are based on semantic vectors that use only existing linguistic resources—contexts and collocations from the Active Dictionary of Russian (AD; Apresjan et al. 2014). All the experiments are performed on 10 Russian verbs, whose senses are taken from AD. We evaluate all methods on contexts sampled from the web-based RuTenTen11 corpus (Kilgarriff et al. 2004). To the best of our knowledge, this is the first evaluation of a semi-supervised word sense disambiguation method for Russian verbs.

## 2. Method

The aim of our method is to be able to perform word sense disambiguation for Russian verbs using only existing linguistic resources, without any additional annotation. Such method needs a predefined sense inventory, and it is convenient if a single resource provides both senses and examples of their usage. In this paper we use sense inventory and examples from the Active Dictionary of Russian, a reliable resource with

a strong theoretical basis in sense distinction that reflects contemporary language (Apresjan et al. 2014). Our disambiguation method consists of two major components: a context representation technique and a classifier trained on labeled contexts (examples and collocations from AD). More precisely, for each sense we extracted all examples (short and common usages), illustrations (longer, full-sentence examples from the Russian National Corpus), collocations, synonyms and analogues. Each example, illustration, etc. was treated as a separate context of a word used in a particular sense. Context representation technique takes contexts (some fixed window of words before and after the disambiguated word) as an input and produces some real-valued vector as an output. This vector is then fed into the classifier, which predicts the sense of a context. Method implementation is available online<sup>1</sup>.

There are a lot of options to choose from when building a context representation: whether to take word order into account, to parse the input sentence, to use lemmatization, to extract morphological features, how to represent words, etc. An important consideration is the amount of training data available, and the nature of the classification task. We have evaluated several options, but the most robust and performant for this task turned out to be representing context as a weighted average of individual word vectors. Word vectors are obtained by training a word2vec (Mikolov et al. 2013) model on a large lemmatized corpus (about 2 billion words—combined RuWac, lib.ru and Russian Wikipedia). Resulting vectors usually have 200–2000 dimensions and represent semantically similar words as vectors with similar directions. Using such vectors as input features allows us to leverage information from large unlabeled corpora and to generalize from one labeled context to all contexts that contain semantically similar words. We take a weighted average of individual word vectors: weights represent to which extent each word affects the sense of a context. Consider for example the word *verbovat* ‘(to recruit)’. If you see words such as *agent* ‘agent’ or *razvedka* ‘intelligence service’ in the context, these words alone give a strong hint about the sense of the target word. We give more weight to words that are more likely to be seen in the context of the target word than on their own ( $q_i$  here is the weight of the word). Negative weights are clipped to 0. If the word is unattested in available contexts, it is given a low weight of 0.2:

$$q_i = \begin{cases} \max\left(0, \ln \frac{P(w_i|c)}{P(w_i)}\right) & \text{if } P(w_i|c) > 0 \\ 0.2 & \text{if } P(w_i|c) = 0 \end{cases}$$

The context vector  $\vec{c}$  is a weighted average of word in vectors, where words are taken from the window of 10 words before and after the target word, crossing sentence boundaries:

$$\vec{c}_n = \sum_{\substack{i=n-10 \\ i \neq n}}^{n+10} q_i \vec{w}_i$$

We have evaluated several classification approaches. In the first approach (Mean-Vec), we calculate an average of all context vectors for each sense during training, thus obtaining a single vector for each sense. When disambiguating an unlabeled

<sup>1</sup> <https://github.com/lopuhin/sensefreq>

context, we find a sense whose vector is the closest to the context vector. In the second approach (LR-Vec) we use an ensemble of logistic regression models with strong regularization. Each model is implemented as neural network that takes context vectors as input, applies dropout with  $p = 0.5$  (Hinton et al. 2012), and a final softmax layer. An ensemble of 5 models is trained using stochastic gradient descent, and prediction of the most confident model is taken as the ensemble output. Dropout is employed to prevent overfitting on a small number of training contexts available from dictionary. The third approach (kNN-Vec) is a  $k$ -nearest neighbours classifier, where the sense is determined by  $k$ -closest neighbours of the input vector, and ties are broken using closeness between sense and context vectors. Best results were obtained with  $k = 3$ . Both Mean-Vec and kNN-Vec use cosine similarity as a closeness metric.

We also used two baseline methods (Bayes and SVM), as they are quite simple and often give strong results, especially with little training data. They used a different context representation: instead of semantic vectors, contexts were represented as bags of words with lemmatization and TF-IDF (term frequency inverse document frequency) downscaling. This means that inputs to these methods are sparse vectors, where each dimension corresponds to a word seen during training, and the only non-zero entries in each context vector correspond to words in this context vector. We compared two robust approaches for classification: a Bayes classifier and a support vector machine (SVM).

### 3. Evaluation

Word sense disambiguation accuracy is evaluated on contexts for 10 Russian verbs randomly sampled from the web-based RuTenTen11<sup>2</sup> corpus, the largest Russian corpus consisting of 18 billion tokens integrated into the Sketch Engine system (Kilgarriff et al. 2004). 7 verbs were randomly chosen polysemous verbs from the first issue of the Active Dictionary of Russian, and 3 (*vosprinimat'*, *brodit'*, *vypolzti*) were specifically chosen to make our test set of words maximally diverse. The word *brodit'* is homonymous (*Ona brodit po ulitse / Kvas brodit*) and both homonyms have several senses. Each word has 100 contexts labeled by one annotator. Annotated corpus is available online<sup>3</sup>.

Comparison of word sense disambiguation accuracy on 10 verbs is presented in Table 1. For each word the table lists the number of senses, the ratio of the most frequent sense (MFS) in training data, and the disambiguation accuracy for 5 approaches. MFS is usually considered a strong baseline in supervised WSD (Navigli 2009), but this is not the case here: dictionary examples are used for training, and it is not known which sense is the most frequent. We see that approaches based on semantic vectors (the first three) perform significantly better, with LR-Vec giving the best average accuracy.

<sup>2</sup> Kutuzov and Kuzmenko (2015), Lopukhina, Lopukhin and Nosyrev (submitted) compared RNC and RuTenTen11 and found that they agree in most cases, including polysemous words.

<sup>3</sup> <https://github.com/lopukhin/ruslang-wsd-labeled>

**Table 1:** Word sense disambiguation accuracy on verbs from RuTenTen11 using Active Dictionary of Russian for training

Word	Senses	MFS	Mean-Vec	LR-Vec	kNN-Vec	Bayes	SVM
<i>atakovat'</i>	3	0.49	0.67	0.58	0.61	0.44	0.39
<i>bayukat'</i>	2	0.74	0.83	0.83	0.78	0.77	0.72
<i>boltat'sya</i>	5	0.37	0.55	0.60	0.47	0.45	0.28
<i>bombardirovat'</i>	4	0.47	0.83	0.83	0.85	0.64	0.55
<i>brodit'</i>	6	0.78	0.65	0.88	0.69	0.83	0.51
<i>verbovat'</i>	3	0.43	0.64	0.61	0.62	0.37	0.50
<i>vlit'</i>	3	0.89	0.91	0.90	0.92	0.64	0.65
<i>volnovat'sya</i>	3	0.96	0.97	0.97	0.97	0.96	0.94
<i>vosprinimat'</i>	3	0.70	0.77	0.78	0.78	0.71	0.65
<i>vypolzti</i>	6	0.42	0.54	0.48	0.42	0.36	0.30
<b>Average</b>	<b>3.8</b>	<b>0.624</b>	<b>0.735</b>	<b>0.745</b>	<b>0.712</b>	<b>0.617</b>	<b>0.548</b>

We also see that both baseline methods perform significantly worse than methods based on dense semantic vectors. This could be due to either the classification method or to the feature representation. In Table 2 we compare best performing semantic vector method (LR-Vec) with Bayes and SVM using two different feature representations (dense vectors and sparse vectors). Results suggest that the main gain is from the feature representation, and that LR method is able to use it more effectively.

**Table 2:** Comparison of average WSD accuracy across methods and feature representations

Method \ Features	Dense vectors	Sparse vectors
<b>LR</b>	<b>0.745</b>	0.600
<b>Bayes</b>	0.682	0.617
<b>SVM</b>	0.693	0.548

Some words in Table 1 have a much lower WSD accuracy: *atakovat'*, *boltat'sya*, *verbovat'* and *vypolzti*. Most of them (except for *atakovat'*) were difficult to the human annotator too, although we have no interrater agreement score to back this up. *Verbovat'* required very long contexts, often exceeding what was available, and general situation understanding. Disambiguation between several senses of *boltat'sya* required geometrical reasoning and sometimes quite specific knowledge. *Vypolzti* required animacy disambiguation, and often coreference resolution with long contexts. On the other hand, *brodit'*, which was by far the hardest word to annotate, has 88% accuracy with LR-Vec approach.

It is possible to analyse how our method determines the sense of the context. This is especially easy for the Mean-Vec approach: each sense has a vector, each word in a disambiguated context also has a vector and a weight, so we can calculate

closeness of a sense vector to each individual word vector. In Table 3 we present contexts for word *atakovat'* (to attack) with 3 senses: 'to attack someone in sport', 'to attack in a war conflict', and 'to ask someone many questions'. Word weight in the second column with values greater than 1.0 is highlighted in bold. The next three columns show similarities of individual words to each sense vector without taking weight into account. Context vector is calculated using individual word vectors and weights, and then compared with sense vectors, but since all operations are linear, it is equivalent to calculating weighted similarities for individual words, as shown in the table. It is interesting to observe that although the words *turki* (Turks) or *CSKA* (a sport team name) are absent in the training data, sense vectors help the method infer that the first word is more likely to be used in the war context, and the second in the sport context, due to their semantic similarity to other words in training data.

**Table 3:** Mean-Vec approach applied to an example context

	Weight	1: sport	2: war	3: questions
<i>v</i>	0.0	—	—	—
<i>pervom</i>	0.2	0.2	0.2	0.0
<b><i>tajme</i></b>	<b>3.0</b>	<b>0.4</b>	0.1	0.1
<i>neskol'ko</i>	0.5	0.1	0.1	0.0
<b><i>opasnyh</i></b>	<b>1.2</b>	<b>0.2</b>	0.1	0.0
<i>momentov.</i>	0.7	0.2	0.1	0.0
<i>Prichem</i>	0.2	0.2	0.1	0.0
<b><i>turki</i></b>	<b>2.3</b>	0.2	<b>0.3</b>	0.1
<i>staralis'</i>	0.2	0.1	0.0	0.0
<i>bol'she</i>	0.2	0.0	0.0	0.0
<i>atakovat'</i>	—	—	—	—
<b><i>levym</i></b>	<b>1.5</b>	<b>0.2</b>	<b>0.2</b>	0.0
<b><i>flangom,</i></b>	<b>3.9</b>	<b>0.5</b>	<b>0.4</b>	0.0
<i>gde</i>	0.0	—	—	—
<i>iz-za</i>	0.3	0.1	0.1	0.0
<i>kadrovyh</i>	0.0	—	—	—
<i>problem</i>	0.0	—	—	—
<i>v</i>	0.0	—	—	—
<b><i>zashhite</i></b>	<b>1.1</b>	0.1	<b>0.2</b>	0.0
<i>u</i>	0.0	—	—	—
<b><i>CSKA</i></b>	<b>1.6</b>	<b>0.4</b>	0.1	0.1
Result		<b>0.7</b>	0.4	0.1

Our method is targeted to be able to train on a relatively low number of diverse examples from the dictionary, but it is possible to evaluate it in a supervised manner. We can train it on 50 out of 100 labeled contexts for each word, using other 50 for testing. WSD accuracy in Table 4 is averaged on 10 random splits of data into training and test sets. All approaches perform better when trained on contexts from corpora

than on dictionary examples, but the gap is smaller than what we observed for nouns in (Lopukhina, Lopukhin and Nosyrev, submitted).

**Table 4:** WSD accuracy with different training data  
(Active Dictionary of Russian and 50 contexts from RuTenTen11 corpus)

Training data	Mean-Vec	LR-Vec	kNN-Vec	Bayes	SVM
Active Dictionary	0.735	<b>0.745</b>	0.712	0.617	0.548
RuTenTen11	0.763	<b>0.778</b>	0.738	0.636	0.682

In the last part of this section we would like to analyze some implementation details and their effect on WSD accuracy (LR-Vec approach accuracy is reported unless otherwise specified):

- **Lemmatization.** It was not clear upfront whether lemmatization is required: on one hand, word2vec models achieve higher quality vectors from lemmatized corpora, on the other, lemmatization loses information that could be especially useful for verbs. It turns out that higher-quality word vectors are more important: WSD accuracy with lemmatization is significantly better: 0.738<sup>4</sup> vs. 0.709 with all other parameters being equal.
- **Stop-words.** Stop-words removal is a common pre-processing step in many NLP applications. But in this case stop-words can be important for disambiguation: for example, for *vosprinimat'* the WSD accuracy with stop-words removed is 0.68, whereas with stop-words included it is 0.78. Words such as *kak* turn out to be important discriminating factors here, while other stop-words are filtered out with weights. Stop-words were not removed when training a word2vec model, either.
- **Context size.** The context size determines which words will be included in the context vector used for classification. Our experiments show that almost all words benefit from larger contexts: average WSD accuracy is 0.745 with context size 10, 0.710 with 5 and 0.687 with 3, where context size 3 means that 3 words before and after the disambiguated word are used. An example of a larger context benefitting disambiguation is the context for *atakovat'* in Table 3: only words *tajm* and *CSKA* have high similarity with the correct sense, and they are far from the disambiguated word.
- **Accounting for word order.** All approaches described so far do not take word order into account, but word order is definitely useful for human speakers. We experimented with building contexts vectors by concatenating individual word vectors and using already described classifiers. But all methods we tried suffered from overfitting on larger context windows, with best results below 0.6 on window size 3. A possible way to resolve this problem is to build context representation that takes word order into account without supervision, and then use this representation to train a disambiguation method. We believe this is a promising direction for future study.

<sup>4</sup> This is different from 0.745 in Table 1 due to use of a smaller corpus and lower-dimensional vectors.



- **Using weights.** Using weight when building context vectors improves WSD accuracy, but less than what we observed for nouns in (Lopukhina, Lopukhin and Nosyrev, submitted): we achieve 0.741 without weights and 0.745 with weights.
- **Word2vec model.** We used the skip-gram model in all experiments. The main hyperparameters are window size and vector dimensionality. Larger window size (10–20) allows capturing topic/domain similarity, while smaller windows (2–3) better capture functional and syntactic similarity. In our experiments moderate window sizes (3–5) performed better: 0.723 with window size 10 and 0.745 with window size 5. Vector dimensionality also turned out to be an important factor: Mean-Vec and kNN-Vec significantly benefited from increasing it, while LR-Vec was less sensitive but still achieved best performance with 2048-sized vectors.

## 4. Conclusions

We presented a method that is able to reach a word sense disambiguation accuracy of 75% for Russian verbs. Our method uses context representation based on semantic vectors with weighting and dictionary information: examples and collocations from the Active Dictionary of Russian (Apresjan et al. 2014). Method’s implementation and labeled contexts are available online on <https://github.com/lopuhin/sensefreq>.

Although our method shows good accuracy on many words, there are some words that are especially problematic. We believe that the general approach with semantic vectors is sound, but our current implementation is basic: it does not explicitly identify the verb’s arguments, relying instead on weights to filter out words that help disambiguation. It would be interesting to check if a more sophisticated approach would improve the method’s accuracy.

We would like to apply our method to other verbs and evaluate it on different types of predicates. The approach with the best performance will be implemented in the model for verb sense frequencies estimation, an ongoing project introduced in (Lopukhina, Lopukhin and Nosyrev, submitted)<sup>5</sup>.

## Acknowledgements

This work was supported by the grant from the Russian Scientific Foundation (No. 16-18-02054). We thank the anonymous reviewers for the many helpful comments and suggestions they made.

---

<sup>5</sup> <http://sensefreq.ruslang.ru>

## References

1. *Agirre et al. 2009* — Agirre E. et al. Knowledge-Based WSD and Specific Domains: Performing Better than Generic Supervised WSD //IJCAI. — 2009. — Pp. 1501–1506.
2. *Apresjan et al. 2014* — Apresjan, Jury D., editor. Active Dictionary of Russian. A-G., Moscow. 2014.
3. *Bartunov et al. 2015* — Bartunov, Sergey, Dmitry Kondrashkin, Anton Osokin, and Dmitry Vetrov. Breaking sticks and ambiguities with adaptive skip-gram. arXiv preprint arXiv:1502.07257. 2015.
4. *Fellbaum 1998* — Fellbaum, Christiane, editor. WordNet, An Electronic Lexical Database. The MIT Press, Cambridge, MA. 1998.
5. *Gries et al. 2010* — Gries S. T., Hampe B. and Schönefeld D. (2010). Converging evidence II: More on the association of verbs and constructions. In: Empirical and experimental methods in cognitive/functional research, CSLI Publications, pages 59–72.
6. *Hanks 2008* — Hanks P. Mapping meaning onto use: a Pattern Dictionary of English Verbs. In Proceedings of the ACL, Utah. 2008.
7. *Hanks and Pustejovsky 2005* — Hanks P. and Pustejovsky J. (2005). A Pattern Dictionary for Natural Language Processing. *Revue Française de linguistique appliquée*, 10(2):63–82.
8. *Hinton et al. 2012* — Hinton G. E. et al. Improving neural networks by preventing co-adaptation of feature detectors. arXiv preprint arXiv:1207.0580. 2012.
9. *Huang 2012* — Huang E. H. et al. Improving word representations via global context and multiple word prototypes //Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1. — Association for Computational Linguistics, 2012. — Pp. 873–882.
10. *Ide and Véronis 1998* — Ide N., Véronis J. Introduction to the special issue on word sense disambiguation: the state of the art // *Computational linguistics*. — 1998. — T. 24. — № 1. — Pp. 2–40.
11. *Kashkin and Lyashevskaya 2013* — Kashkin E. V., Lyashevskaya O. N. Semantic roles and construction net in russian FrameBank. *Computational Linguistics and Intellectual Technologies: Proceedings of the International Conference “Dialogue 2013”* [Komp’yuternaya Lingvistika i Intellektual’nye Tekhnologii: Trudy Mezhdunarodnoy Konferentsii “Dialog 2013”], Bekasovo, pp. 827–836.
12. *Kilgarriff et al. 2004* — Kilgarriff, Adam, Pavel Rychly, Pavel Smrz, and David Tugwell. The Sketch Engine. Information Technology Research Institute Technical Report Series: pp. 105–116. 2004.
13. *Kobritsov et al. 2005* — Kobritsov B. P., Lyashevskaya O. N., Shemanayeva O. N. Lexico-semantic disambiguation in news and newspaper texts: surface filters and statistical estimation. [Snyatie leksiko-semanticheskoy omonimii v novostnyx i gazetno-zhurnal’nyx tekstax: poverxnostnye fil’try i statisticheskaya ocenka]. *Internet-mathematics 2005: automated web-data processing*. M.: 2005. Pp. 38–57.
14. *Kobritsov et al. 2007* — Kobritsov B. P., Lyashevskaya O. N., Toldova S. Ju. Semantic disambiguation of verbs using government patterns extracted from explanatory dictionaries. [Snyatie semanticheskoy mnogoznachnosti glagolov

- s ispolzovaniem modelej upravleniya, izvlechennyx iz e-lektronnyx tolkovyx slovarej]. Internet-mathematics — 2007, electronic publication, <http://download.yandex.ru/IMAT2007/kobricov.pdf>. — 2007.
15. *Kustova and Toldova 2008* — Kustova G. I., Toldova S. Ju. RNC: semantic filters for verb sense disambiguation. [NKRYa: semanticheskie fil'try dlya razresheniya mnogoznachnosti glagolov]. Russian National Corpus. — 2006. — T. 2008. — Pp. 258–276.
  16. *Kustova G., Toldova S.* RNC: Semantic filters for the verb disambiguation [‘NKRJA: semanticheskiye filtry dlja razresheniya mnogoznachnosti glagolov’]. Russian national corpus: 2006–2008. New results and perspectives. [‘Natsionalnyi korpus russkogo jazyka: 2006–2008. Novye rezultaty i perspektivy’]. Saint Petersburg: Nestor-Istorija. 2009.
  17. *Kutuzov and Kuzmenko 2015* — Kutuzov, Andrey, and Elizaveta Kuzmenko. Comparing Neural Lexical Models of a Classic National Corpus and a Web Corpus: The Case for Russian. Computational Linguistics and Intelligent Text Processing. Springer International Publishing: pp. 47–58. 2015.
  18. *Lopukhina, Lopukhin and Nosyrev, submitted* — Lopukhina A., Lopukhin K. and Nosyrev G. Automated word sense frequency estimation for Russian nouns. Submitted to Quantitative Approaches to the Russian Language (available online: [http://sensefreq.ruslang.ru/download/Automated\\_Word\\_Sense\\_Frequency\\_Estimation\\_for\\_Russian\\_Nouns\\_Lopukhina\\_et\\_al.pdf](http://sensefreq.ruslang.ru/download/Automated_Word_Sense_Frequency_Estimation_for_Russian_Nouns_Lopukhina_et_al.pdf))
  19. *Loukachevitch and Chujko 2007* — Loukachevitch N. V., Chujko D. S. Automated lexical disambiguation using thesauri. [Avtomaticheskoe razreshenie leksicheskoy mnogoznachnosti na baze tezaurusnyx znaniy]. Internet-mathematics 2007. Ekaterinburg: 2007. Pp. 108–117.
  20. *Loukachevitch and Dobrov 2007* — Loukachevitch N. V., Dobrov B. V. Lexical disambiguation based on domain-specific thesaurus. Computational Linguistics and Intellectual Technologies: Proceedings of the International Conference “Dialogue 2007” [Komp’yuternaya Lingvistika i Intellektual’nye Tekhnologii: Trudy Mezhdunarodnoy Konferentsii “Dialog 2007”], Bekasovo, pp. 400–407.
  21. *Lyashevskaya 2012* — Lyashevskaya O. N. Dictionary of Valencies Meets Corpus Annotation: A Case of Russian FrameBank // Proceedings of EURALEX 15 (7–11 August 2012, Oslo, Norway). Oslo : Oslo University, 2012.
  22. *Mihalcea 2011* — Mihalcea R. Word sense disambiguation // Encyclopedia of Machine Learning. — Springer US, 2011. — Pp. 1027–1030.
  23. *Mikolov et al. 2013* — Mikolov T., Chen K., Corrado G., and Dean J. Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781. 2013.
  24. *Mitrofanova et al. 2008* — Mitrofanova O. A., Panicheva P. V., Lyashevskaya O. N. Statistical word sense disambiguation in contexts for names of physical objects. Computational Linguistics and Intellectual Technologies: Proceedings of the International Conference “Dialogue 2008” [Komp’yuternaya Lingvistika i Intellektual’nye Tekhnologii: Trudy Mezhdunarodnoy Konferentsii “Dialog 2008”], Bekasovo, pp. 358–376.
  25. *Navigli 2009* — Navigli, Roberto. Word sense disambiguation: A survey. ACM Computing Surveys (CSUR) 41.2: 10. 2009.

26. *Neelakantan 2014* — Neelakantan, Arvind, Jeevan Shankar, Alexandre Passos, and Andrew McCallum. Efficient non-parametric estimation of multiple embeddings per word in vector space. Conference on Empirical Methods in Natural Language Processing (EMNLP). 2014.
27. *Ponzetto and Navigli 2010* — Ponzetto S. P., Navigli R. Knowledge-rich word sense disambiguation rivaling supervised systems //Proceedings of the 48th annual meeting of the association for computational linguistics. — Association for Computational Linguistics. — 2010, pp. 1522–1531.
28. *Pradhan et al. 2007* — Pradhan, Sameer S., Edward Loper, Dmitriy Dligach, and Martha Palmer. SemEval-2007 task 17: English lexical sample, SRL and all words. Proceedings of the 4th International Workshop on Semantic Evaluations. Association for Computational Linguistics. 2007.
29. *Reznikova 2014* — Reznikova T. I. Author's style is decent, it really delivers: non-standart semantic shifts in verbs (a study on internet texts). [Slog u avtora neplox, real'no dostavlyaet: o nestandartnyx semanticheskix perexodax v glagol'noj leksike (po dannym interneta)]. Modern Russian in the internet. M: 2014, pp. 169–181.
30. *Rozina 2005* — Rozina R. I. Semantic development in Russian literature and modern slang. [Semanticheskoe razvitie slova v russkom literaturnom yazyke i sovremennom slenge]. M.: Azbukovnik, 2005.
31. *Schutze 1998* — Schütze, Hinrich. 1998. Automatic word sense discrimination. Computational Linguistics 24, 1: pp. 97–124.
32. *Shemanayeva et al. 2007* — Shemanayeva O. Ju., Kustova G. I., Lyashevskaya O. N., Rakhilina E. V. Semantic filters for the word sense disambiguation in RNC: adjectives. Computational Linguistics and Intellectual Technologies: Proceedings of the International Conference “Dialogue 2007” [Komp'yuternaya Lingvistika i Intellektual'nye Tekhnologii: Trudy Mezhdunarodnoy Konferentsii “Dialog 2007”], Bekasovo, pp. 582–588.