Computational Linguistics and Intellectual Technologies:
Proceedings of the International Conference “Dialogue 2017”

Moscow, May 31—June 3, 2017

SEMANTIC ROLE LABELING WITH NEURAL NETWORKS FOR TEXTS IN RUSSIAN

Shelmanov A. O. (shelmanov@isa.ru),
Devyatkin D. A. (devyatkin@isa.ru)

Federal Research Center “Computer Science and Control” of Russian Academy of Sciences, Moscow, Russia

We present and evaluate neural network models for semantic role labeling of texts in Russian. The benchmark for evaluation and training was prepared on the basis of the FrameBank corpus. The paper addresses different aspects of learning a neural network model for semantic role labeling on different feature sets including syntactic features acquired with the help of SyntaxNet. In this work, we rely on architecture engineering and atomic features instead of commonly used feature engineering. We investigate the ability of learning a model for labeling arguments of “unknown” predicates that are not present in a training set using word embeddings as features for the replacement of predicate lemmas. We publish the prepared benchmark and the models. The experimental results can be used as a baseline for further research in semantic role labeling of texts in Russian.

Keywords: semantic parsing, semantic role labeling, frame parsing, neural network, word embeddings, deep learning

1. Introduction

Semantic role labeling (SRL) is a useful type of linguistic analysis that maps varying low-level syntactic representations of sentences to more abstract argument-predicate structures. Predicates in these structures are words that express situations, they are verbs, verbal nouns, and verb forms. Arguments are words and phrases (often noun phrases) that play a role in a situation expressed by a predicate. These semantic roles capture meaning of arguments and explicitly present meaningful aspects encoded in the sentence by an author. The significance of semantic role annotation lies in the fact that such abstract semantic representations naturally can be applied for a variety of natural language processing tasks, which require comparison of texts by their meaning: question answering [Shen and Lapata, 2007], information extraction [Christensen et al., 2011], information search [Osipov et al., 2014], machine translation [Liu and Gildea, 2010], and others.

The majority of the state-of-the-art methods for SRL rely on supervised learning techniques that require a lot of annotated data. This is a problem for developing a good SRL system, since creating such an annotated resource is a very expensive and difficult task. Such resources have been created for several languages. For today, the most used and researched resources are FrameNet [Baker et al., 1998] and Propbank [Kingsbury and Palmer, 2002]—corpora that provide SRL annotations for English texts. For long time, there was no such a resource for Russian. Although several semantic parsers that produce SRL-like annotations were presented in the past, they mostly relied on hand-crafted rules and dictionaries [Sokirko, 2001], as well as on training on automatically annotated corpus [Shelmanov and Smirnov, 2014]. However, the recent release of FrameBank corpus [Lyashevskaya, 2012; Lyashevskaya and Kashkin, 2015] enables new capabilities of using machine learning techniques for creating semantic role labelers that work with Russian language and for new fundamental research in this direction. In this work, we investigate the ability of training a semantic role labeler based on neural networks using various types of linguistic features and word embeddings [Le and Mikolov, 2014].

The FrameBank provides the hierarchical role schema, the lexicon with predicates that mostly are verbs (and verb forms), and the partially annotated text corpus for more than 800 predicates. We note that the verb coverage by examples of the corpus is still not very high. This encourages us to develop semi-supervised approach to improving the parser capabilities of annotating sentences with "unknown" predicates that are not present in the training set. Therefore, in addition of creating and evaluating neural network models for SRL we also investigate the ability of using word embeddings to mitigate the problem of low verb coverage.

The main contributions of this paper are the following:

1. The openly available benchmark for evaluation of semantic parsers for Russian language based on FrameBank corpus[1].
2. The openly-available neural network models for semantic role labeling trained on FrameBank and evaluated on different feature sets.
3. The method for processing "unknown" predicates based on word embeddings.

2. Related Work

One of the first methods for SRL presented in [Gildea and Jurafsky, 2002] was based on a simple statistical model. Since then, more sophisticated machine learning techniques have been elaborated very quickly. Several shared tasks CoNLL-2004, 2005, 2008, and 2009 [Hajic et al., 2009] set up some common benchmarks and revealed useful machine learning approaches, in which authors investigated different features sets, task decomposition methods, and global inference techniques. Early works devoted to SRL heavily relied on complex feature engineering. The advances in neural network training as well as in learning of meaningful representations of words sparked new interest to problem of SRL. In many recent works, researchers propose new neural network approaches based on architecture engineering. It was

[1] http://nlp.isa.ru/framebank_parser

revealed that neural networks do not need complex features, instead they can rely on atomic features or even on very low-level representations like tokens or n-grams. Such models often significantly outperform the traditional ones. In the rest of the section, we review the recent works devoted to SRL for English and Russian.

One of the first well-known publications, in which feature engineering was replaced by an architecture engineering, is [Collobert et al., 2011]. The researchers presented and applied a single neural network model to various natural language processing tasks including part-of-speech tagging, named entity recognition, and semantic role labeling. They showed that this approach allows to reduce domain and task specific feature engineering. The main idea of this work lies in exploiting latent interactions between features in big and mostly unlabeled training sets.

The paper [Roth and Lapata, 2016] proposes a novel model for SRL based on recurrent neural network. The researchers claim that complex syntactic structures are not analyzed well by baseline approaches. They proposed a model that processes subsequences of lexicalized dependency paths and learns suitable embedding representations of them. The researchers empirically showed that such embeddings can improve results over the previous baseline SRL approaches.

In the similar way, [FitzGerald et al., 2015] presented a new model for SRL, in which arguments and semantic roles are jointly embedded in a shared vector space for a given predicate. This model utilizes finer-grained semantic similarity between roles. The researchers trained a neural network to approximate the potential functions of a graphical model designed for the SRL task and used this network to build embeddings. They showed that the proposed model can learn jointly from PropBank and FrameNet to achieve performance improvements on the smaller FrameNet dataset.

In [Foland and Martin, 2015], authors proposed a method for SRL based on convolutional and time-domain neural networks. The method takes into account features derived from a dependency parser output. The authors explored the benefits of adding increasingly more complex dependency-based features to the model. The proposed method demonstrated state-of-the-art performance and low computational requirements.

Recently, several works proposed end-to-end SRL approaches that do not require syntactic features. These approaches allow to avoid losing information between different stages of text processing.

In [Marcheggiani et al., 2017], researchers proposed a simple syntax-agnostic model for dependency-based SRL. That model predicts predicate-argument dependencies relying on states of a bidirectional LSTM encoder [Hochreiter and Schmidhuber, 1997]. The authors showed that sufficient accuracy on English texts can be achieved even without syntactic information using only local inference. It was also approved that the model is more robust on the standard out-of-domain test set than the baselines.

Similar approach was proposed in [Zhou and Xu, 2015]. The researchers applied a model based on bidirectional recurrent network for end-to-end SRL. They did not use any syntactic information but relied only on original text as the input features. The model was evaluated on SRL task of CoNLL-2005 and coreference resolution task of CoNLL-2012. It outperformed the previous state-of-the-art ensemble models. The authors revealed that the proposed model is better at processing longer sentences than the baseline approaches.

It is also worth noting great interest to joint modeling of syntax and semantics in many works devoted to SRL. For example, in [Swayamdipta et al., 2016], a transition-based model for SRL that jointly produces syntactic and semantic dependencies was presented. The model is based on a stack of LSTM cells and is used for representation of the entire algorithm state. The researchers also proposed a greedy inference algorithm, which works in linear time. They obtained the best published parsing performance among models that jointly learn syntax and semantics on the CoNLL-2008, 2009 datasets.

There are a few works devoted to semantic role labeling of Russian language texts. In the previous work, we used rule and dictionary-based semantic parser for creating automatically annotated corpus for training a model for SRL [Shelmanov and Smirnov, 2014]. In [Kuznetsov, 2015; Kuznetsov, 2016], SVM-based semantic role labeler was trained on FrameBank corpus. The corpus was supplemented by syntactic features generated by the pipeline presented in [Sharoff and Nivre, 2011]. The author also performed clustering of lexis features to extract additional semantic information from the corpus and used ILP-optimization approach for post processing. This work is based on the pre-release version of the FrameBank corpus and does not provide the tools for the data preparation, modeling, and evaluation. In this work, author did not use neural networks and word embeddings as features mostly relying on feature engineering.

In our work, instead of feature engineering, we use atomic features with word embeddings and neural networks. We also research the problem of semantic role labeling for "unknown" predicates (out-of-domain predicates) and propose the simple approach to that problem. We publish the benchmark for model construction and evaluation on the FrameBank corpus.

3. Neural Network Models for FrameBank Parsing

We present two neural network models for semantic role labeling. These models mostly diverge in the way different feature types are aggregated. We used the following features:

Categorical:

1) Various types of morphological features of both an argument and a predicate: part of speech, grammar case, animacy, verb form, time, passiveness, and others. ("morph").
2) Relative position of an argument in a sentence with respect to a predicate. ("rel_pos").
3) Predicate lemma ("pred_lemma").
4) Preposition of an argument extracted from a syntax tree ("arg_prep").
5) Name of a syntax link from argument to its parent in a syntax tree ("synt_link").

Embeddings:

1) Embedding of an argument lemma ("arg_emeddings").
2) Embedding of a predicate lemma ("pred_emeddings").

The first neural network model has the simple architecture that acquires all features of an argument: sparse and dense, as a single vector and propagates them through three dense layers. The two hidden layers have ReLU activations and the output layer has softmax activation. The softmax activation is a standard way of producing final probabilities of classes in a multinomial classification task. The ReLU activation is a rectifier function that propagates only positive signal through a network. This activation function is convenient since it simplifies training of deep architectures and results in lesser overfitting effect than many other functions. In the hidden layers, we use batch normalization [Ioffe and Szegedy, 2015]. In this technique, inputs of layers are normalized in each mini-batch, which drastically increases the training speed of networks and also regularizes them. The network also has two dropout layers for additional regularization. We will refer to this model as "simple".

The second neural network is intended to handle embeddings and categorical features more intelligently than the "simple" one. The problem of processing the both types of features lies in their different nature. The categorical features are sparse, therefore, merging them with embeddings within one dense layer would result in a big number of parameters. The better way of handling this case is to embed sparse categorical features first and merge them later. Therefore, the complex model has the same types of layers but the first layer is split into several chunks: a chunk for categorical features, a chunk for an argument embedding (if it is present in a feature set), and a chunk for a predicate embedding (if it is present in a feature set). Such an architecture is much smaller than the "simple" one in terms of parameters, thus, it overfits less and is trained faster. We will refer to this model as "complex". The Figure 1 depicts the neural network architectures.

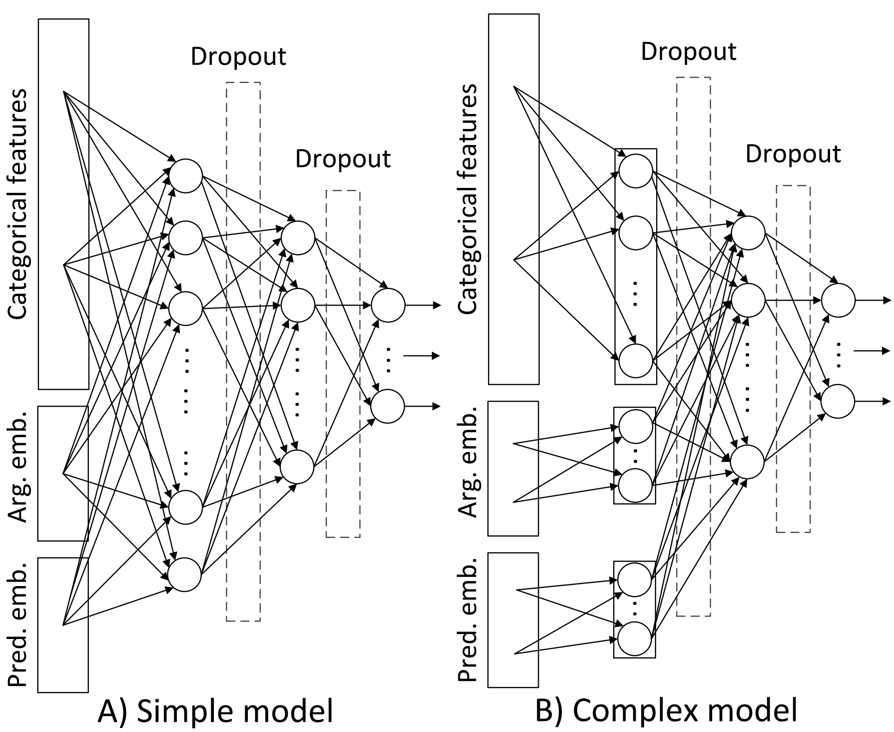

Figure 1. Architectures of neural network models

We compile these models with Adam optimizer [Kingma and Ba, 2015] and a standard categorical cross entropy loss function. These models and different subsets of the aforementioned features are used for labeling of arguments of "known" predicates.

For labeling arguments of "unknown" predicates, we also use the similar architectures. However, in this setting we cannot rely on predicate lemma feature, since there will be no lemma in the test set known by the model. In this setting, predicate embeddings should give the most significant impact on a network performance. Embeddings, due to the way they are built, encode semantic similarities of words in a low dimensional vector space [Le and Mikolov, 2014]. Many text processing methods that have been recently developed rely heavily on this remarkable property of embeddings and demonstrate its great usefulness. We investigate the ability of substituting predicate lemma feature in SRL parser by its embedding. Embeddings are built in an unsupervised manner on a huge unlabeled corpus, so model does not need to see every predicate lemma in a small semantically labeled training set to obtain its embedding. Since such embeddings encode similarities between words, they could also encode similarities between frame structures of predicates. Therefore, we can use training examples of "known" predicates to infer the frame structure of "unknown" predicates that are similar to the former in an embedding vector space. The bigger the similarity, the more precise we can restore frame structures of "unknown" predicates.

In the setting for "unknown" predicates, we additionally used early stopping in the training procedure since it becomes useless to tune fixed number of epochs for out-of-domain test set.

4. Experiments

4.1. Experiment Setup

We used the publicly released version of FrameBank corpus[2]. The corpus contains annotated text examples that consist of multiple sentences. Tokens in the sentences are annotated with morphological and some other features. The role and the predicate annotations are separated from the texts. The original version of the corpus does not contain explicit exact mapping between role annotations and tokens or text spans. To mitigate this problem, we created the automatic tool for mapping predicates and arguments with core roles to text tokens.

To create the syntax annotation for FrameBank, we used Google's SyntaxNet parser[3] [Andor et al., 2016]. This parser was trained on SyntagRus treebank [Nivre et al., 2008] and provides high quality parsing for Russian texts according to [Alberti et al., 2017]. We used dockerized version of SyntaxNet with a model for Russian[4,5].

[2] https://github.com/olesar/framebank

[3] https://github.com/tensorflow/models/tree/master/syntaxnet

[4] https://github.com/IINemo/docker-syntaxnet_rus

[5] https://hub.docker.com/r/inemo/syntaxnet_rus

The parser creates a fully connected dependency tree for a sentence with syntax tags on every parent-child link. The syntax structure corresponds to well-known Universal dependencies format [Nivre et al., 2016].

After the mapping procedure, we obtained the corpus that contains examples for 803 predicates. We selected the subcorpus by keeping only predicates that have at least 10 examples. This results in 572 predicates left in the subcorpus. We also filtered out arguments with infrequent semantic roles and preprocessed erroneous role labels that do not correspond to the role ontology of FrameBank published in [Kashkin and Lyashevskaya, 2013]. The final version of the whole experimental dataset contains 53,151 examples with 44 different semantic roles.

The word embeddings used in our experiments are provided by RusVectores 2.0[6] [Kutuzov and Andreev, 2015]. They were pre-trained on Russian national corpus and have 300 dimensions. We note high quality of the model; however, we also note that a large portion of predicates (verbs) presented in FrameBank are not covered by it. Therefore, more than 17,000 examples in our dataset have zero predicate embeddings.

The hyperparameters of the proposed neural network models on different features sets were tuned using the greedy strategy. We mostly tuned dropout ratio, the size of internal dense layers, and a number of training epochs.

For the simple baseline, we use a parser that assigns the most frequent semantic role to every argument in the test set. Obviously, this baseline has low performance, but it shows the skewness in the evaluation set, which reflects the complexity of the task and the impact of other models.

We evaluated our models using macro and micro F_1 score. We note that our results are not directly comparable with the results presented in [Kuznetsov, 2015]. This is due to the fact that the author used different annotation scheme and different pre-release version of FrameBank corpus with unknown preprocessing procedures.

4.2. Evaluating Models on "Known" Predicates

In the first experiment, we evaluate our models on different feature sets: lexis, morphological, syntactic, and word embeddings. In each feature set we also use relative position feature. The performance of the models is assessed using five-fold cross validation on the selected subcorpus of FrameBank. The evaluation results are presented in Table 1.

Table 1. Performance of the models on different feature sets

Model + feature set	Macro F_1-score, %	Micro F_1-score, %
Baseline	0.5 ± 0.0	11.6 ± 0.2
Simple + morph	22.8 ± 0.6	35.4 ± 0.3
Simple + morph + pred_lemma	71.2 ± 0.6	76.1 ± 0.5
Simple + morph + pred_emeddings	62.0 ± 0.4	65.2 ± 0.3

6 http://rusvectores.org/en

Model + feature set	**Macro F_1-score, %**	**Micro F_1-score, %**
Simple + morph + pred_lemma + arg_prep	75.9 ± 0.4	79.2 ± 0.2
Simple + morph + pred_lemma + arg_prep + synt_link	76.8 ± 0.5	80.3 ± 0.3
Simple + morph + pred_lemma + arg_prep + synt_link + arg_embeddings + pred_embeddings	78.6 ± 0.4	81.8 ± 0.2
Complex + morph + synt + pred_lemma + arg_embeddings + pred_embeddings	**79.2 ± 0.3**	**82.3 ± 0.2**

Since the evaluation dataset is not very unbalanced, the baseline that marks the dominant class has a very low performance. Adding morphological features of predicates and arguments results in a substantial improvment over the baseline: Δmicro $F_1 = 23.8\%$. This setting shows the importance of low-level linguistic features in semantic role labeling without appealing to any semantics of arguments and predicates. This performance could be achieved without any knowledge about meaning of predicates or arguments and syntactic information. With adding predicate lemmas, we drastically improve performance of labeling by Δmicro F_1=40.7%, which is not surprising. Since frame structures are invoked by a predicate that represents a situation, roles can be very specific to predicates. Without knowledge of which predicate invoked the current frame, in many cases, it is impossible to distinguish roles of morphologically similar arguments. The results of the setting, in which we substitute predicate lemma with its embeddings, show that the performance drop without predicate lemmas is not very big, when at least embeddings of predicates are present. This enables the ability of building a model for "unknown" predicates relying on properties of word embeddings.

In the next setting, the feature set is composed of morphological features, predicate lemmas, and argument preposition. The preposition in Russian is considered to be very important for semantic role labeling. We observe an Δmicro $F_1 = 3.1\%$ increase compared to the model that does not take it into account, which is very significant for building a good semantic parser. Adding names of parent syntax links of arguments as features extends this improvement by another percent. We used only basic syntactic features: preposition and the parent link, whereas it is also worth adding, e.g., the syntactic path from argument to predicate as suggested in many previous works. We leave this for the future work, since it would require comparison of many different embedding techniques for a very sparse space of syntactic paths. We also note that although the syntactic features are important for building a good SRL model, they do not drastically increase the performance of the parser. Following several techniques presented in related work that suggest syntax agnostic models for English, we consider the task of creation an accurate model for Russian without appealing to syntactic parsing also feasible.

Adding embeddings of arguments and predicates to the rest of the features yields the best results. The "simple" model as expected gives the smallest performance gain Δmicro $F_1 = 1.5\%$. Adding embeddings directly as additional dimensions results in a big growth of a number of parameters. Therefore, such a network tends to overfitting. The "complex" model due to its architecture is twice as smaller in terms of parameters compared to the "simple" one. It gives another small but significant performance improvement Δmicro $F_1 = 0.5\%$ compared to the "simple" model on the full feature set. It also trains and runs much faster than the "simple" model.

4.3. Evaluating Models on "Unknown" Predicates

In the second experiment, we research the importance of word embeddings in the task of labeling arguments of "unknown" predicates. For this setting, we split the selected subcorpus of FrameBank in two parts: training and testing in such a way that the part for testing contains only predicates that are absent from the part for training. We perform evaluations for two different split methods. In the first split, the test part is composed from examples for predicates that have highly similar predicates in the training part. For that, cosine similarity of every two predicate embeddings is calculated. The top 27 similar pairs of predicates are distributed into different parts of corpus. In this case, we get 49,709 training and 3,442 testing examples. Such a split represents the good case, in which semantic similarity of "unknown" predicates to "known" ones can be captured by their word embeddings. This case should be easy for the models. In the second split, in a contrary, we compose the test part from predicates that are least similar to any of the "known" predicates. This split yields 50,093 training examples, and 3,058 testing examples with 21 predicates in the test set. This case should be the hardest for the models to handle. In this experiment, we do not perform cross-validation. Instead, we train models five times with different random seeds and test them on the prepared holdout. This does not prevent overfitting but alleviates the problem of randomness of model training.

We compare "simple" model with all categorical features and without embeddings, the "complex" model with categorical features and only argument embeddings, and the "complex" model with categorical features, as well as argument and predicate embeddings. The evaluation results are presented in Table 2 and 3.

Table 2. Evaluation of the models on the "unknown" predicates in the "good" split

Model + feature set	Macro F_1-score, %	Micro F_1-score, %
Baseline	0.4	9.6
Simple	13.7 ± 0.4	24.6 ± 0.3
Complex + arg_embeddings	19.4 ± 0.3	31.9 ± 0.5
Complex + arg_pred_embeddings	41.4 ± 0.7	66.7 ± 1.1

Table 3. Evaluation of the models on the "unknown" predicates in the "bad" split

Model + feature set	Macro F_1-score, %	Micro F_1-score, %
Baseline	0.7	13.2
Simple	9.1 ± 0.2	24.8 ± 0.5
Complex + arg_embeddings	14.5 ± 0.7	27.2 ± 0.1
Complex + arg_pred_embeddings	24.1 ± 1.5	41.4 ± 2.2

The results show that there is a substantial performance drop in macro and micro scores on the "unknown" predicates. However, we see that on completely unseen predicates the complex model with embeddings shows a decent micro score. The model on the good split shows expectedly better results than on the bad split. This confirms the significance of the presence in the training set of predicates that are similar to "unknown" ones in the embedding vector space. However, we note that even on a "bad" split the model with embeddings shows much better performance compared to the "simple" model that uses only morphological and syntactic categorical features. We should also note again that the substantial part of predicate embeddings in the training set are zeros due to already mentioned limitations of used language model. This definitely affects the performance of the SRL models. In the future work, it is worth training neural networks using more complete language models.

5. Conclusion and Future Work

We presented the neural network models for semantic role labeling of Russian texts. We also presented the basic benchmark based on FrameBank corpus for evaluation of parsers for SRL. Both the models and the benchmark are openly available[7]. The proposed models were evaluated on different feature sets. The achieved scores could be used as a baseline for the future research. We also investigated the method for training a labeler for arguments of "unknown" predicates using word embeddings. We demonstrate that good embeddings are essential for building a model for "unknown" predicates, however, it is not enough to approach the performance of models trained and tested on in-domain data.

In this work, we did not provide the semantic argument identifier and did not perform the global inference step in the SRL parser. The reason for that consists in the fact that FrameBank corpus provides very sparse annotations (not every argument in sentences is labeled). Therefore, learning inference procedure using straightforward approach is hardly possible. However, in the future work, we are looking forward to adapt self-learning techniques on the partially annotated data and use integer linear programming inference that does not require additional training to further boost the performance of the parser.

Acknowledgments

The project is supported by the Russian Foundation for Basic Research, project number: 16-37-00425 "mol_a".

References

1. *Alberti C., Andor D., Bogatyy I., Collins M., Gillick D., Kong L., et al.* (2017), SyntaxNet Models for the CoNLL 2017 Shared Task. arXiv preprint arXiv:1703.04929.
2. *Andor D., Alberti C., Weiss D., Severyn A., Presta A., Ganchev K., Petrov S., and Collins M.* (2016), Globally normalized transition-based neural networks. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 2442–2452.

---

[7] http://nlp.isa.ru/framebank_parser

3. *Baker C. F., Fillmore C. J., and Lowe J. B.* (1998), The Berkeley FrameNet project. In Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics-Volume 1, pp. 86–90.
4. *Christensen J., Mausam, Soderland S., and Etzioni O.* (2011), An analysis of open information extraction based on semantic role labeling. In Proceedings of the sixth international conference on Knowledge capture, pp. 113–120.
5. *Collobert R., Weston J., Bottou L., Karlen M., Kavukcuoglu K., and Kuksa P.* (2011), Natural language processing (almost) from scratch. Journal of Machine Learning Research, 12(Aug):2493–2537.
6. *FitzGerald N., Täckström O., Ganchev K., and Das D.* (2015), Semantic role labeling with neural network factors. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 960–970.
7. *Foland W. and Martin J.* (2015), Dependency-based semantic role labeling using convolutional neural networks. In Proceedings of the Fourth Joint Conference on Lexical and Computa-tional Semantics, pp. 279–288.
8. *Gildea D. and Jurafsky D.* (2002), Automatic labeling of semantic roles. Computational linguistics, 28(3):245–288.
9. *Hajic J., Ciaramita M., Johansson R., Kawahara D., Mart M. A., Màrquez L., Meyers A., Nivre J., Padó S., Štepánek J., et al.* (2009), The CoNLL-2009 shared task: Syntactic and semantic dependencies in multiple languages. In Proceedings of the Thirteenth Conference on Computational Natural Language Learning: Shared Task, pp. 1–18.
10. *Hochreiter S. and Schmidhuber J.* (1997), Long short-term memory. Neural computation, 9(8):1735–1780.
11. *Ioffe S. and Szegedy C.* (2015), Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Proceedings of the 32nd International Conference on Machine Learning (ICML-15), pp. 448–456.
12. *Kashkin E. and Lyashevskaya O.* (2013), Semantic roles and construction net in Russian FrameBank [semanticheskie roli i set'konstrukcij v sisteme framebank]. In Computational Linguistics and Intellectual Technologies. Papers from the Annual International Conference "Dialogue" (2014), volume 1, pp. 297–311, (in Russian).
13. *Kingma D. and Ba J.* (2015), Adam: A method for stochastic optimization. In Proceedings of the International Conference on Learning Representations (ICLR).
14. *Kingsbury P. and Palmer M.* (2002), From treebank to PropBank. In Proceedings of the International Conference on Language Resources and Evaluation, pp. 1989–1993.
15. *Kutuzov A. and Andreev I.* (2015), Texts in, meaning out: Neural language models in semantic similarity tasks for Russian, volume 2, pp. 133–144.
16. *Kuznetsov I.* (2015), Semantic role labeling for Russian language based on Russian FrameBank. In Proceedings of International Conference on Analysis of Images, Social Networks and Texts, pp. 333–338.
17. *Kuznetsov I.* (2016), Automatic semantic role labelling in Russian language [Avtomaticheskaya razmetka semanticheskih roley v russkom iazike], PhD thesis (in Russian).
18. *Le Q. V. and Mikolov T.* (2014), Distributed representations of sentences and documents. In Proceedings of the International Conference on Machine Learning, volume 14, pp. 1188–1196.

19. *Liu D. and Gildea D.* (2010), Semantic role features for machine translation. In Proceedings of the 23rd International Conference on Computational Linguistics, pp. 716–724.
20. *Lyashevskaya O.* (2012), Dictionary of valencies meets corpus annotation: a case of Russian FrameBank. In Proceedings of the 15th EURALEX International Congress, volume 15.
21. *Lyashevskaya O. and Kashkin E.* (2015), FrameBank: a database of Russian lexical constructions. In International Conference on Analysis of Images, Social Networks and Texts, pp. 350–360.
22. *Marcheggiani D., Frolov A., and Titov I.* (2017), A simple and accurate syntax-agnostic neural model for dependency-based semantic role labeling. arXiv preprint arXiv:1701.02593.
23. *Nivre J., Boguslavsky I. M., and Iomdin L. L.* (2008), Parsing the SynTagRus treebank of Russian. In Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1, pp. 641–648.
24. *Nivre J., de Marneffe M.-C., Ginter F., Goldberg Y., Hajic J., Manning C. D., McDonald R., Petrov S., Pyysalo S., Silveira N., et al.* (2016), Universal dependencies v1: A multilingual treebank collection. In Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016), pp. 1659–1666.
25. *Osipov G., Smirnov I., Tikhomirov I., Sochenkov I., Shelmanov A., and Shvets A.* (2014), Information retrieval for R&D support. In Professional Search in the Modern World, pp. 45–69.
26. *Roth M. and Lapata M.* (2016), Neural semantic role labeling with dependency path embeddings. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 1192–1202.
27. *Sharoff S. and Nivre J.* (2011), The proper place of men and machines in language technology: Processing Russian without any linguistic knowledge. In Computational Linguistics and Intellectual Technologies. Papers from the Annual International Conference "Dialogue" 2011.
28. *Shelmanov A. O. and Smirnov I. V.* (2014), Methods for semantic role labeling of Russian texts. In Computational Linguistics and Intellectual Technologies. Papers from the Annual International Conference "Dialogue" (2014), number 13, pp. 607–620.
29. *Shen D. and Lapata M.* (2007), Using semantic roles to improve question answering. In Proceedings of the Conference on Empirical Methods on Natural Language Processing and Computational Natural Language Learning, pp. 12–21.
30. *Sokirko A.* (2001), A short description of Dialing Project, available at URL: http://www.aot.ru/docs/sokirko/sokirko-candid-eng.html
31. *Swayamdipta S., Ballesteros M., Dyer C., and Smith N. A.* (2016), Greedy, joint syntactic-semantic parsing with stack LSTMs. In Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning, CoNLL 2016, pp. 187–197.
32. *Zhou J. and Xu W.* (2015), End-to-end learning of semantic role labeling using recurrent neural networks. In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pp. 1127–1137.