Записки научных
семинаров ПОМИ
Том 499, 2021 г.

R. B. Galinsky, A. M. Alekseev, S. I. Nikolenko

IMPROVING NEURAL MODELS FOR NATURAL LANGUAGE PROCESSING IN RUSSIAN WITH SYNONYMS

Abstract. Large-scale deep learning models, including models for natural language processing, require large datasets for training that could be unavailable for low-resource languages or for special domains. We consider a way to approach the problem of poor variability and small size of available data for training NLP models based on augmenting the data with synonyms. We design a novel augmentation scheme that includes replacing words with synonyms and reshuffling the words, apply it to the Russian language, and report improved results for the sentiment analysis task.

§1. Introduction

Large-scale neural network models usually require very large datasets to be trained efficiently. While it is usually easy to collect large unlabeled text datasets, it may be hard to collect large datasets for a specific problem such sentiment analysis, syntactic parsing, machine translation, and so on. One possible way in such cases is to take advantage of transfer-learning-like approaches, reusing the knowledge about the domain field obtainable from other datasets, e.g. [17], or domain-specific multi-task approaches (please refer to [38]). In this paper, we will focus on a different way to approach the problem of poor variability and small size of the available data for training the natural language processing models.

In computer vision, it is a common practice to augment the input datasets by certain changes in the input images. Computer vision yields itself very easily to such modifications: if we slightly crop, shift, or contract an image, change lighting conditions or downsample to reduce resolution, the objects on the image will remain the same, and the recognition target

Key words and phrases: Deep learning, natural language processing, data augmentation, sentiment analysis.
This research was supported by the St. Petersburg State University, research project "Artificial Intelligence and Data Science: Theory, Technology, Industrial and Interdisciplinary Research and Applications".

can be reused. This is not even denoising as in denoising autoencoders, it is simply new training data obtained with little efforts. These augmentation procedures are used in most modern computer vision models; please refer to, e.g., [25, 37] and references therein. As of 2019, special software simplifying the augmented images generation is developed with the source code open, e.g. [5, 18]

However, in natural language processing one cannot simply change a word at random and assume that the "big picture" will remain exactly the same. Ideally, we might use human paraphrases but they are impossible to obtain in the necessary quantities. Zhang et al. [46] propose a straightforward idea for such data augmentation: use a human-generated standard thesaurus (WordNet [11, 35] in their case) and replace some words at random with their direct synonyms. They report improved results with this augmentation. It appears that there might be other transformations helpful for textual data augmentation, and this problem may warrant further study.

In this work, we modify and apply this scheme to the Russian language. Besides, we propose and evaluate another data augmentation scheme based on extending user reviews (in a sentiment analysis task) with additional adjectives. For other approaches to different tasks in Russian language processing utilizing different dictionaries and thesauri please refer to the works [27, 40, 42].

The paper is organized as follows. In Section 2, we discuss an important idea for natural language processing based on deep learning, namely moving from word-level embeddings such as *word2vec* to character-level models. Section 3 discusses in detail the data augmentation procedures we have evaluated. Section 4 shows experimental results that validate that augmentation based on synonyms does improve sentiment analysis results, and Section 5 concludes the paper.

This work is an extended version of the paper "Improving Neural Network Models for Natural Language Processing in Russian with Synonyms" that was presented on the AINL 2016 conference.

§2. Character-level models and data augmentation in natural language processing

Distributed word representations are models that map each word occurring in the dictionary to a Euclidean space [13]. The modern field of word embeddings started with the work [3], subsequently extended in [4].

Extending previous work on statistical language models that were usually based on word n-grams [7, 8, 14, 21], Bengio et al. proposed the idea of *distributed word representations*, the idea of word embeddings was applied back to language modeling, e.g., in [32,33,36], and then, starting from the works of Mikolov et al. [31, 34], word representations have been applied for numerous natural language processing problems, including text classification, extraction of sentiment lexicons, part-of-speech tagging, syntactic parsing and so on.

Soon, *character-level representations* appeared that take into account the actual characters that comprise a word. First attempts at this problem involved decomposing a word into *morphemes*, the smallest units of meaning in written language [6,28,41], but emphasis quickly shifted to the characters themselves. In [26], Ling et al. present a *character to word* (C2W) model for learning word embeddings based on bidirectional LSTMs [15,16]. Recent work on character-level models for morphologically rich languages has introduced morphological smoothing that could model the morphological variation in the word embedding space [10] and explicit representations of morphological features for reinflection [19].

Character-level models are especially important for developing NLP models for the Russian language for two main reasons. First, they are very well suited for languages with rich morphology, such as Russian; Russian contains plenty of words that are tightly linked with each other (have the same root), and shades of meaning are distinguished with morphemes. It would be obviously very wasteful to treat all of them as separate words. One can use available morphological analyzers to connect different forms of the same word (we do so in auxiliary steps of this work too), but then one has to either disregard morphological data, which loses meaningful information, or again treat different forms of a word as different words. Second, character-level models are also well suited for studies of user-generated texts such as user reviews, social network statuses, blog posts, and the like; user-generated texts abound with typos, intentional misspellings (and other noise in general; also addressed in [29,30]), word spelling variations, and so on, which are immediately recognized by human readers but are impossible to pick up for a word-based model.

In this work, we concentrate on a specific form of *data augmentation* for natural language processing, i.e., extending the dataset by adding randomized variation to training examples while keeping the target variable unchanged. While data augmentation is a key technique in, e.g., computer

vision, where it is used basically universally, using data augmentation and synthetic data in natural language processing has been a much less developed field.

Still, there have been works that use data augmentation for NLP. One approach is to simply drop out certain words [39]. A development of this idea shown in [44] switches out certain words, replacing them neither with zeros (as dropout does) nor with synonyms but with random words from the vocabulary. The work [45] develops methods of data noising for language models, adding noise to word counts in a way reminiscent of smoothing in language models based on n-grams.

As the closest to our work, we highlight [46], which used direct data augmentation with synonyms. The work [43], which concentrated on studying tweets, proposed to use embedding-based data augmentation, using neighboring words in the word vector space as synonyms. The work [22] extends the augmentation with synonyms approach by replacing words in sentences with other words in paradigmatic relations with the original words, as predicted by a bi-directional language model at the word positions.

In our experiments, we used a character-level model similar to the one presented in [46], where Zhang et al. develop a natural approach to constructing character-level representations based on convolutional neural networks. They report significant improvements for standard text classification problems. They also suggest a straightforward way for data augmentation: replacing a word with its direct synonym. However, for Russian and other morphologically rich languages this scheme is harder to apply as the new word has to match the syntax as well as the semantics of the old word. We are not aware of previous work on such data augmentation for Russian; other data augmentation approaches have included, e.g., anaphora resolution as a preprocessing technique to improve the word embeddings [24].

§3. Data augmentation approaches

3.1. Replacing words with their synonyms. To achieve data augmentation with synonyms, we begin with collecting and filtering a set of pairs of synonymous words. We begin with publicly available dictionaries of synonyms for the Russian language, collected from online versions of

dictionaries of synonyms [1] [1] [2] [2]. We also used a general frequency vocabulary of the Russian language, running a preliminary filter to exclude archaic or very rare words.

At the data augmentation stage, we use an explicit morphological analyzer *pymorphy* [23]; naturally, the use of an automated analyzer introduces a certain share of errors but the errors are rare enough to still lead to overall improvement. First, we use *pymorphy* to find the part of speech and other morphological data for all words and leave only nouns and adjectives. Then we take the synonyms to have the same gender: masculine noun with masculine noun and so on.

In dictionaries, it often happens that some words are more general, and others are their special cases. In linguistics, this is known as *hyponymy*: a hyponym is a word of phrase whose semantic meaning is included within the meaning of a more general word, which is called a *hyperonym*. In this case, it may be incorrect to replace the general word with a more specific, less abstract word. For example, it is almost always correct to replace *car* with *automobile* but not with *minivan*, although a dictionary may mark *car* as a synonym for *minivan*. Having not used thesauri containing the corresponding relations, one cannot determine which word in an asymmetrical pair of synonyms is more general, so as the next filter the reflexivity was checked: we only use w_1 as a synonym for w_2 if both w_1 is marked as a synonym for w_2 in the dictionary and w_2 is marked as a synonym of w_1 in the dictionary.

At this point, we have a set S of unordered pairs of synonyms that we assume to be safe to use for replacement.

Next, we go through the input text and process in with *pymorphy*. The analyzer outputs morphological features for each word. For every word w, we run the following procedure:

• remember its morphological features and take its base form w_0 as suggested by *pymorphy*;

• look for the synonyms of the base form w_0 in the set of synonyms S, getting the set of synonyms $S_w = \{w' \mid (w_0, w') \in W\}$;

• sample a synonym w'_0 from S_w according to a multinomial distribution with probabilities proportional to the word frequencies (overall frequencies in the Russian language).

[1] https://nlpub.ru/Словарь_Абрамова; http://speakrus.ru/dict/#abramov

[2] http://publ.lib.ru/ARCHIVES/A/ALEKSANDROVA_Zinaida_Evgen'evna/_Aleksandrova_Z.E..html

Note that at the sampling stage, we can either include the word w_0 itself in S_w, regarding it as its own synonym, or leave it out. Our experiments show that it is beneficial to include the word w_0 itself in S_w, sometimes leaving the word in place even if it does have synonyms in S. This turns out to be important in cases when the word is very frequent, and synonyms are rare and unlikely to appear so it is better to leave it in place.

Then we use *pymorphy* to map the word w'_0 back to the form used in the review and replace the original word w with the resulting form w'.

3.2. Reshuffling the words. Another straightforward technique for data augmentation is to reshuffle the words. The correct way to shuffle words would be to automatically construct parse trees from the sentences and then randomly change places of certain subtrees; the less rigid word order in Russian makes this approach attractive. However, in this work we only use a very simple and obviously incorrect approach of word reshuffling, basically turning it into a bag of words. Somewhat surprisingly, we will demonstrate in Section 4 that even if we shuffle all words randomly, the resulting sentiment recognition quality does not change all that much.

3.3. Adding new adjectives. Experiments with reshuffling words in a review (we did not get significant reduction in quality from basically converting the review into a bag of words) suggest that we could try to generate "simulated reviews" by simply sampling suitable words. We tested this idea with an experiment on adding new adjectives and/or verbs since adjectives and verbs are usually the most characteristic words for sentiment evaluation (as our counting experiments shown below suggest).

For the new augmentation procedure, we have chosen to add new adjectives. For preprocessing, we collected the following statistics, again using *pymorphy* for part of speech tagging and lemmatization:

• count how many times a given (lemmatized) adjective occurs in the dataset both in positive and negative reviews (some of these results are discussed below and shown in Table 2);

• count how many times a given adjective appears before or after a noun (we did not perform full syntactic parsing here, simply counted occurrences of noun-adjective and adjective-noun bigrams);

• count how many times a given adjective occurs next to a given noun.

After these statistics have been connected, for the augmentation we go over the text of a given review, looking for nouns. If a noun w does not

have an associated adjective (i.e., an adjective either before or after it), we perform the following procedure:

• sample whether to add an adjective to this noun based on statistics on how often w appears with and without adjectives;

• if an adjective should be added, sample which one to add from the multinomial distribution with probabilities proportional to the numbers of times different adjectives occur in positive and negative reviews next to this noun;

• then sample whether it should be added before or after the noun based on the corresponding statistic;

• then add the resulting adjective to the text.

After this augmentation procedure, we get reviews with additional adjectives that adhere to the dataset statistics and do indeed most often "make sense" for the corresponding words.

Table 1. Dataset statistics

Dataset	Reviews		
	Positive	Negative	Total
Basic: torg + Restoclub	63088	35046	98134
Augmented with adjectives	126176	70092	196268
Augmented with synonyms	125523	69849	195372
Test dataset: TripAdvisor	26807	11075	37882

§4. Evaluation

4.1. Datasets and basic statistics. For experimental evaluation, we have chosen the sentiment analysis problem since it is relatively easy to mine large train and test datasets for this classical NLP problem. To try to train for general sentiment rather than for a specific subject domain, we have collected our dataset (referenced further as an "original" one) from two very different sources: marketplace reviews from *torg.mail.ru* and restaurant reviews from *www.restoclub.ru* (referenced further as torg and Restoclub, respectively). The basic statistics are shown in Table 1.

Next, we have applied the augmentation procedures described in detail in Section 3 to obtain two extended datasets: one augmented with additional adjectives as shown in Section 3.3 and another augmented with direct synonyms as shown in Section 3.1. In each case, we have extended

Table 2. Imbalanced words in various parts of speech

Word		**Counts**				
Russian	English	Pos.	% pos.	Neg.	% neg.	Diff.
Adjectives						
замечательный	wonderful	5537	0.088	1153	0.033	0.055
огромный	huge	7251	0.115	2052	0.059	0.056
вежливый	polite	6853	0.109	1759	0.050	0.058
красивый	pretty	7921	0.126	2331	0.067	0.059
прекрасный	beautiful	6713	0.106	1620	0.046	0.060
...	...					
должный	must	3171	0.050	5249	0.150	-0.100
отвратительный	disgusting	332	0.005	2716	0.077	-0.072
ужасный	terrible	453	0.007	2746	0.078	-0.071
никакой	bad	4060	0.064	4616	0.132	-0.067
данный	this	3229	0.051	4111	0.117	-0.066
Nouns						
свадьба	wedding	4718	0.075	1244	0.035	0.039
атмосфера	atmosphere	6734	0.107	2317	0.066	0.041
площадь	area	4937	0.078	1153	0.033	0.045
храм	temple	4271	0.068	363	0.010	0.057
собор	cathedral	5045	0.080	599	0.017	0.063
...	...					
итог	total	3710	0.059	6349	0.181	-0.122
счёт	bill	3374	0.053	6063	0.173	-0.120
ответ	response	1846	0.029	5047	0.144	-0.115
том	volume	6017	0.095	7109	0.203	-0.107
фильм	movie	5461	0.087	6773	0.193	-0.107
Verbs						
помочь	help	3561	0.056	1245	0.036	0.021
отмечать	note	3133	0.050	975	0.028	0.022
посетить	visit	5801	0.092	2165	0.062	0.030
порадовать	gladden	5406	0.086	1546	0.044	0.042
доставить	deliver	5939	0.094	1804	0.051	0.043
...	...					
звонить	call	2111	0.033	5017	0.143	-0.110
вернуть	return	1345	0.021	4562	0.130	-0.109
стать	become	5450	0.086	6725	0.192	-0.106
позвонить	call	5148	0.082	6223	0.178	-0.096
говорить	speak	3078	0.049	4791	0.137	-0.088
Adverbs						
вовремя	timely	2966	0.047	550	0.016	0.031
удобно	conveniently	3787	0.060	977	0.028	0.032
отлично	excellently	4599	0.073	1065	0.030	0.043
приятно	pleasantly	7743	0.123	1834	0.052	0.070
обязательно	certainly	6939	0.110	1172	0.033	0.077
...	...					
вообще	generally	5386	0.085	7373	0.210	-0.125
потом	after	4395	0.070	5830	0.166	-0.097
почему	why	3022	0.048	5058	0.144	-0.096
более	more	5848	0.093	5982	0.171	-0.078
видимо	seemingly	1836	0.029	3725	0.106	-0.077

Table 3. Experimental results

Dataset	Best accuracy	
	Test set	TripAdvisor set
Orig. dataset: torg + Restoclub	0.8457	0.7163
Orig. with reshuffled words	0.8445	0.7160
Augmented with adjectives	0.7241	0.5430
Augmented with synonyms	0.8700	0.7020

the original dataset by approximately a factor of two, adding one modified review for each original one.

Besides, to test how well the resulting sentiment models transfer to a different domain, we have collected another, smaller dataset from a completely different source: hotel reviews from the *TripAdvisor* Web site. This dataset was never used in training, but we evaluated the quality of our models on it. Note, however, that results on the *TripAdvisor* dataset are expected to be significantly worse not only because the domain is different but also due to the properties of the *TripAdvisor* dataset itself: it has a different distribution of review scores, with about 90% of the reviews scoring five stars.

Another interesting piece of data is the number of occurrences of words in positive and negative reviews; in our experiments, it plays a role for data augmentation with adjectives and verbs as discussed in Section 3.3. Table 2 shows the most imbalanced positive and negative words for various parts of speech; some entries represent lemmatization errors or confusion between different words but mostly they paint a reasonable picture. It is also clear that the most imbalanced (colored) words are adjectives and nouns.

4.2. Training the model. Our model was based on the *keras* [9] implementation of the model presented in [46] (`fhttps://github.com/johnb30/py_crepe`). We have used the same topology: starting from character quantization with a simple 1-of-m encoding, the unprocessed text data is fed to a convolutional net with 6 convolutional layers, 3 fully connected layers, and 2 dropout modules between fully connected layers for regularization; we used 1024 units on the fully connected top layers. We have used the Adam optimizer [20] for training. All experiments were conducted on a

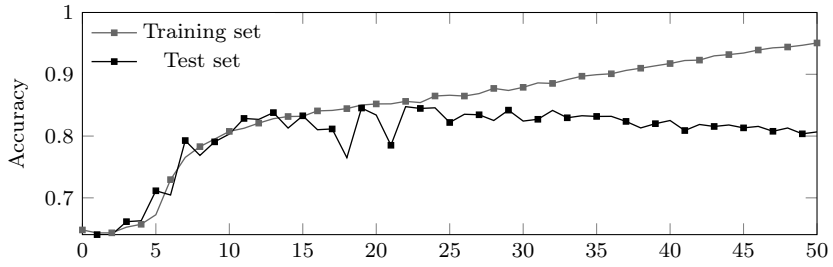
single NVIDIA Titan X GPU. The training and test set errors for the original dataset are shown on Figure 1a.

4.3. Word reshuffling. In this experiment, we have trained and applied the model to the original dataset with all words in each review randomly shuffled. Somewhat surprisingly, test set accuracy of the resulting model is virtually indistinguishable from the original (best test scores being 0.8445 and 0.8457, respectively), and the score on a separate TripAdvisor dataset from a completely different domain (please refer to the Table 3) is also approximately the same as the original model. This indicates that, first, it might make sense to add new words to reviews even if they slightly violate grammatical rules because the grammar does not seem to matter much; and second, that the models still have a long way to go before they can achieve real understanding of sentiment since it does obviously depend on word order.

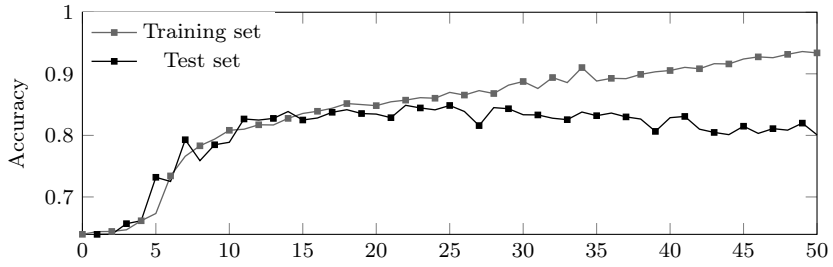
4.4. Augmented datasets. We have also trained and tested the model on augmented datasets, with synonyms and with additional adjectives. Figure 1c shows training and test errors for the dataset augmented with synonyms, Figure 1d, for the dataset augmented with adjectives, Figure 2 compares the test set errors across all four experiments, and Table 3 summarizes the results.

The results on augmentation with synonyms were positive: we have seen significant improvements in accuracy in our experiments: approximately +0.025 compared to the results on the unmodified dataset. However, data augmentation with additional adjectives did not work (test accuracy 0.7241), producing worse results than even the original dataset (test accuracy 0.8457), while at the same time the accuracy on the training set was rising much faster and higher than for other augmentation strategies (compare Fig. 1d with Fig. 1a-c). This can be explained by overfitting: adding sentiment-heavy adjectives has resulted in a training set full with specific words that mark sentiment, so the model had trained to recognize these words and could not process the test set without this abundance quite as well.

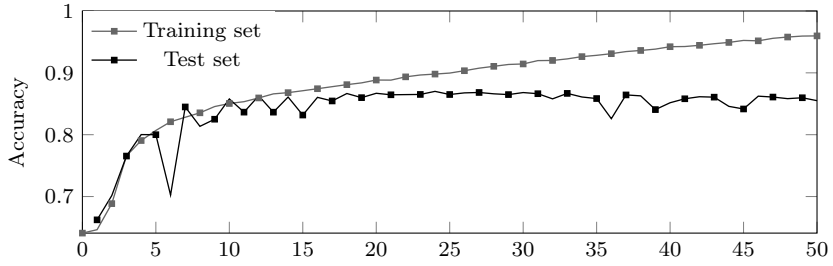
4.5. TripAdvisor experiment. We have also performed an additional experiment, evaluating the quality of the resulting models on a problem domain where they had not been trained, namely on hotel reviews from TripAdvisor. The accuracy of different models on this additional dataset is



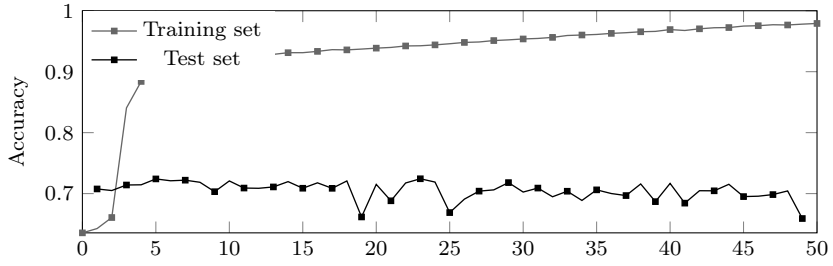
(a) Original dataset: torg + Restoclub



(b) Dataset with randomly reshuffled words



(c) Dataset augmented with synonyms



(d) Dataset extended with adjectives

Figure 1. Accuracy on training and test sets; the X-axis shows training epochs.

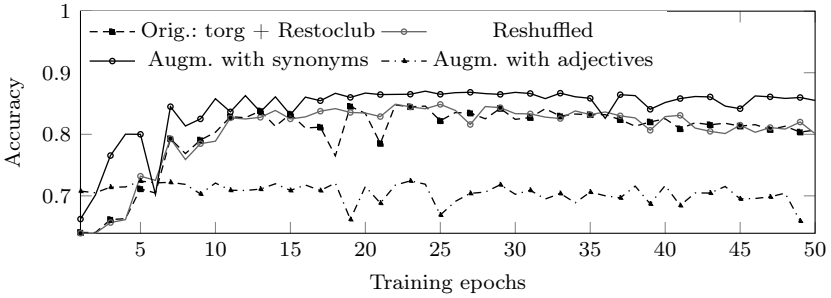


Figure 2. A comparison of test set accuracies of all models in the study

also shown in Table 3. The results indicate that so far, the resulting sentiment models do not transfer easily from one domain to another: across all datasets, results on the test set are significantly worse, and the improvements from synonym-based data augmentation have disappeared. This indicates that general-purpose sentiment models are still subject for further work.

§5. Conclusion

In this work, we have introduced and evaluated different approaches to data augmentation for natural language processing in the context of character-level predictive models for Russian language. Our results show promise: it appears that even simple data augmentation with synonyms taken from the well-known word lists can yield significant improvements for the sentiment analysis task. We propose to use augmentation with synonyms as a tool to extend insufficiently large datasets. On the other hand, we have seen that not every augmentation method is beneficial: an extension with extra adjectives turned out to produce worse results.

In further work, we plan to improve upon these augmentation approaches and produce state of the art character-level models for the Russian language. We also believe that for further work, it will be interesting to combine different approaches to augmentation and find out which combinations of augmentation techniques are beneficial. Another interesting direction would be to try augmentation on smaller datasets and tasks, where augmentation techniques should probably shine even more.

References

1. N. Abramov, *Dictionary of russian synonyms and synonymous phrases*, Moscow: Russkie Slovari, 1999.
2. Z. E. Alexandrova, *Dictionary of russian synonyms*, Moscow: Russkii Yazyk, 2001.
3. Y. Bengio, R. Ducharme, P. Vincent, *A neural probabilistic language model*, **3** (2003), 1137–1155.
4. Y. Bengio, H. Schwenk, J.-S. Senécal, F. Morin, J.-L. Gauvain, *Neural probabilistic language models*, Innovations in Machine Learning, Springer, 2006, pp. 137–186.
5. M. D. Bloice, C. Stocker, A. Holzinger, *Augmentor: an image augmentation library for machine learning*, arXiv preprint arXiv:1708.04680 (2017).
6. J.A. Botha, P. Blunsom, *Compositional morphology for word representations and language modelling*, Proc. 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014, 2014, pp. 1899–1907.
7. P. F. Brown, P. V. deSouza, R. L. Mercer, V. J. D. Pietra, J. C. Lai, *Class-based n-gram models of natural language*, Comput. Linguist. **18** (1992), no. 4, 467–479.
8. S. F. Chen, J. Goodman, *An empirical study of smoothing techniques for language modeling*, Proc. 34th Annual Meeting on Association for Computational Linguistics (Stroudsburg, PA, USA), ACL '96, Association for Computational Linguistics, 1996, pp. 310–318.
9. F. Chollet, *Keras*, `https://github.com/fchollet/keras`, 2015.
10. R. Cotterell, H. Schütze, J. Eisner, *Morphological smoothing and extrapolation of word embeddings*, Proc. 54th Annual Meeting of the ACL, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers, 2016.
11. C. Fellbaum (ed.), *WordNet: an electronic lexical database*, MIT Press, 1998.
12. R. Galinsky, A. Alekseev, S. I. Nikolenko, *Improving neural network models for natural language processing in russian with synonyms*, Proc. 5th conference on Artificial Intelligence and Natural Language, 2016, pp. 45–51.
13. Y. Goldberg, *A primer on neural network models for natural language processing*, CoRR **abs/1510.00726** (2015).
14. J. T. Goodman, *A bit of progress in language modeling*, Comput. Speech Lang. **15** (2001), no. 4, 403–434.
15. A. Graves, S. Fernández, J. Schmidhuber, *Bidirectional LSTM networks for improved phoneme classification and recognition*, Artificial Neural Networks: Formal Models and Their Applications - ICANN 2005, 15th International Conference, Warsaw, Poland, September 11-15, 2005, Proceedings, Part II, 2005, pp. 799–804.
16. A. Graves, J. Schmidhuber, *Framewise phoneme classification with bidirectional LSTM and other neural network architectures*, Neural Networks **18** (2005), no. 5-6, 602–610.
17. J. Howard, S. Ruder, *Universal language model fine-tuning for text classification*, Proc. 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), vol. 1, 2018, pp. 328–339.

18. A. B. Jung, *imgaug*, `https://github.com/aleju/imgaug`, 2018, [Online; accessed 30-Dec-2018].
19. K. Kann, H. Schütze, *Single-model encoder-decoder with explicit morphological representation for reinflection*, Proc. 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 2: Short Papers, 2016.
20. D.P. Kingma and J. Ba, *Adam: A method for stochastic optimization*, CoRR **abs/1412.6980** (2014).
21. R. Kneser, H. Ney, *Improved backing-off for m-gram language modeling*, Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., 1995 International Conference on, vol. 1, 1995, pp. 181–184 vol.1.
22. S. Kobayashi, *Contextual augmentation: Data augmentation by words with paradigmatic relations*, Proc. 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers) (New Orleans, Louisiana), Association for Computational Linguistics, 2018, pp. 452–457.
23. M. Korobov, *Morphological analyzer and generator for russian and ukrainian languages*, Analysis of Images, Social Networks and Texts (M. Yu. Khachay, N. Konstantinova, A. Panchenko, D.I. Ignatov, and V.G. Labunets, eds.), Communications in Computer and Information Science, vol. 542, Springer International Publishing, 2015, pp. 320–332 (English).
24. O. Kozlowa, A. Kutuzov, *Improving distributional semantic models using anaphora resolution during linguistic preprocessing*, Proceedings of International Conference on Computational Linguistics "Dialogue 2016", 2016.
25. Y. LeCun, K. Kavukcuoglu, C. Farabet, *Convolutional networks and applications in vision*, International Symposium on Circuits and Systems (ISCAS 2010), May 30 - June 2, 2010, Paris, France, 2010, pp. 253–256.
26. W. Ling, C. Dyer, A. W. Black, I. Trancoso, R. Fermandez, S. Amir, L. Marujo, T. Luis, *Finding function in form: Compositional character models for open vocabulary word representation*, Proc. 2015 Conference on Empirical Methods in Natural Language Processing (Lisbon, Portugal), Association for Computational Linguistics, 2015, pp. 1520–1530.
27. N. Loukachevitch, M. Nokel, K. Ivanov, *Combining thesaurus knowledge and probabilistic topic models*, International Conference on Analysis of Images, Social Networks and Texts, Springer, 2017, pp. 59–71.
28. M.-T. Luong, R. Socher, C. D. Manning, *Better word representations with recursive neural networks for morphology*, CoNLL (Sofia, Bulgaria), 2013.
29. V. Malykh, *Robust word vectors for russian language*, Proceedings of Artificial Intelligence and Natural Language AINL FRUCT 2016 Conference, Saint-Petersburg, Russia, 2016, pp. 10–12.
30. V. Malykh, *Generalizable architecture for robust word vectors tested by noisy paraphrases*, Proc. of The 6th International Conference On Analysis Of Images, Social Networks, and Texts (AIST), 2017.

31. T. Mikolov, K. Chen, G. Corrado, J. Dean, *Efficient estimation of word representations in vector space*, CoRR **abs/1301.3781** (2013).
32. T. Mikolov, M. Karafiát, L. Burget, J. Cernockỳ, and S. Khudanpur, *Recurrent neural network based language model*, INTERSPEECH **2** (2010), 3.
33. T. Mikolov, S. Kombrink, L. Burget, J. H. Černockỳ, S. Khudanpur, *Extensions of recurrent neural network language model*, Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on, IEEE, 2011, pp. 5528–5531.
34. T. Mikolov, I. Sutskever, K. Chen, G. Corrado, J. Dean, *Distributed representations of words and phrases and their compositionality*, CoRR **abs/1310.4546** (2013).
35. G. A. Miller, *Wordnet: a lexical database for english*, Communications of the ACM **38** (1995), no. 11, 39–41.
36. A. Mnih, G. E. Hinton, *A scalable hierarchical distributed language model*, Advances in neural information processing systems, 2009, pp. 1081–1088.
37. M. Ranzato, G. E. Hinton, Y. LeCun, *Guest editorial: Deep learning*, International Journal of Computer Vision **113** (2015), no. 1, 1–2.
38. S. Ruder, *An overview of multi-task learning in deep neural networks*, arXiv preprint arXiv:1706.05098 (2017).
39. R. Sennrich, B. Haddow, A. Birch, *Edinburgh neural machine translation systems for WMT 16*, Proc. First Conference on Machine Translation: Vol. 2, Shared Task Papers, ACL 2016, pp. 371–376 .
40. V. Solovyev, V. Ivanov, *Knowledge-driven event extraction in russian: corpus-based linguistic resources*, Computational intelligence and neuroscience **2016** (2016), 16.
41. R. Soricut, F. Och, *Unsupervised morphology induction using word embeddings*, Proc. 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Denver, Colorado), ACL, 2015, pp. 1627–1637.
42. E. Tutubalina, S. Nikolenko, *Constructing aspect-based sentiment lexicons with topic modeling*, International Conference on Analysis of Images, Social Networks and Texts, Springer, 2016, pp. 208–220.
43. W. Y. Wang, D. Yang, *That's so annoying!!!: A lexical and frame-semantic embedding based data augmentation approach to automatic categorization of annoying behaviors using #petpeeve tweets*, Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (Lisbon, Portugal), ACL, 2015, pp. 2557–2563.
44. X. Wang, H. Pham, Z. Dai, G. Neubig, *SwitchOut: an efficient data augmentation algorithm for neural machine translation*, Proc. 2018 Conference on Empirical Methods in Natural Language Processing, ACL, 2018, pp. 856–861.
45. Z. Xie, S. I. Wang, J. Li, D. Lévy, A. Nie, D. Jurafsky, A. Y. Ng, *Data noising as smoothing in neural network language models*, 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings, OpenReview.net, 2017.

46. X. Zhang, J. Zhao, Y. LeCun, *Character-level convolutional networks for text classification*, Advances in Neural Information Processing Systems 28 (C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, eds.), Curran Associates, Inc., 2015, pp. 649–657.

E-mail: galinskyifmo@gmail.com

Поступило 2 октября 2020 г.

St. Petersburg State University
7/9 Universitetskaya nab.,
St. Petersburg, 199034 Russia;
St. Petersburg Department of
Steklov Institute of Mathematics,
St. Petersburg, Russia

E-mail: anton.m.alexeyev@gmail.com

St. Petersburg State University
7/9 Universitetskaya nab.,
St. Petersburg, 199034 Russia;
St. Petersburg Department of
Steklov Institute of Mathematics,
St. Petersburg, Russia

E-mail: s.nikolenko@spbu.ru, sergey@logic.pdmi.ras.ru