Computational Linguistics and Intellectual Technologies:
Proceedings of the International Conference “Dialogue 2019”

Moscow, May 29—June 1, 2019

LANGUAGE MODEL EMBEDDINGS IMPROVE SENTIMENT ANALYSIS IN RUSSIAN

Baymurzina D. R. (dilyara.rimovna@gmail.com),
Kuznetsov D. P. (kuznetsov.den.p@gmail.com),
Burtsev M. S. (burtsev.m@gmail.com)

Neural Networks and Deep Learning Lab, Moscow Institute of Physics and Technology, Moscow, Russia

Sentiment analysis is one of the most popular natural language processing tasks. In this paper we introduce pre-trained Russian language models which are used to extract embeddings (ELMo) to improve accuracy for classification of short conversational texts. The first language model was trained on Russian Twitter dataset containing 102 million sentences, while two others were trained on 57.5 million sentences of Russian News and 23.9 million sentences of Russian Wikipedia articles. Although classifiers trained on top of language models perform better than in the case of utilizing of fastText embeddings of the same language style, we show that domain of language model also has a significant impact on accuracy. This paper establishes state-of-the-art results for RuSentiment dataset improving weighted F1-score from 72.8 to 78.5. All our models are available online as well as the source code which allows everyone to apply them or fine-tune on domain-specific data.

Key words: ELMo, embeddings from language model, text classification, sentiment analysis, Russian language

ПОВЫШЕНИЕ КАЧЕСТВА АНАЛИЗА ТОНАЛЬНОСТИ НА РУССКОМ ЯЗЫКЕ С ИСПОЛЬЗОВАНИЕМ ВЕКТОРНЫХ ПРЕДСТАВЛЕНИЙ ЯЗЫКОВЫХ МОДЕЛЕЙ

Баймурзина Д. Р. (dilyara.rimovna@gmail.com),
Кузнецов Д. П. (kuznetsov.den.p@gmail.com),
Бурцев М. С. (burtsev.m@gmail.com)

Лаборатория нейронных систем и глубокого обучения, Московский физико-технический институт (национальный исследовательский университет), Москва, Россия

Анализ тональности является одной из наиболее популярных задач обработки естественного языка. В данной работе мы представляем предобученные русские языковые модели, которые используются для получения векторных представлений слов при решении задачи классификации разговорных текстов. Одна языковая модель обучена на 102 миллионах предложений русского Twitter, а две другие — на 57,5 миллионах предложений русских новостей и 23,9 миллионах предложений из русских статьей Wikipedia. Несмотря на то, что классификаторы, обученные на векторных представлениях, извлеченных из языковых моделей, показывают результаты лучше, чем те, что обучены на векторных представлениях fastText соответствующего языкового стиля, мы показываем, что домен языковой модели также оказывает значительное влияние на качество классификации. В данной работе достигается новое наилучшее качество для набора данных RuSentiment, повышающее предыдущий результат с 72,8 значения взвешенной F1-метрики до 78,5. Все представленные модели и исходный код, в том числе для дообучения языковых моделей, доступны онлайн.

Ключевые слова: ELMo, embeddings from language model, классификация текстов, анализ тональности, русский язык

1. Introduction

Sentiment classification is an important part of chat-bots, from question answering helper on web-site to personal assistant that should track owner's mood and desires. The reason of the statement is that conversation with chat-bot should gratify a user but strongly in accordance to a situation.

There are three basic approaches to sentiment classification task: rule-based solution, machine learning (ML) models and neural networks (NN). Rule-based approach is the most popular because it does not require labelled datasets but only sentiment dictionaries. However, rule-based models often do not take into account context wider than two or three tokens. If it is possible to collect and annotate a domain-specific

dataset, one can use supervised ML or NN models. While ML models are usually build upon embeddings of full text sample obtained from TF-IDF or count vectorizers, NN models assume character or token vector representations. Token embeddings could be obtained via many different methods including bag-of-words, GloVe [15], fastText [1]. However, token embeddings extracted from language models are becoming more and more popular. Language model embeddings allow to perform better even on small task-specific datasets which are often encountered in production.

Embeddings from Language Models (ELMo) [17] are vectors derived from bidirectional LSTM trained to solve the task of language modelling on a large text corpus. ELMo representations are deep and context-dependent. Internal states of the model can be combined and used similarly to other token embeddings like fastText but representation of each word is being formed by left and right context of this word. Language models require large text corpora and significant computational resources to be trained.

We have explored several discussions in Russian NLP community about actual performance of ELMo, and faced a lot of negative responses about accuracy of neural models based on ELMo. Therefore, the paper has two main goals: first of all, we introduce three Russian language models pre-trained on Wikipedia articles, news and twits, and the second one is to compare performance of fastText and ELMo embeddings trained on corpora with different language styles. We demonstrate how the domain of language model influences on the accuracy of a classifier trained over obtained embeddings. Also we introduce the source code which allows to simply fine-tune ELMo on the domain specific data.

2. Related Work

A lack of studies on Russian sentiment analysis is caused by a lack of appropriate datasets. First of all, the largest sentiment lexicon is RuSentiLex [11] which latest version is dated by 2017 although neologisms appear regularly by borrowing from other languages or from positive and negative happenings in political, social and cultural life of Russia.

There are three common datasets for Russian sentiment analysis in academic research: aspect-oriented SentiRuEval 2015 [10], SentiRuEval 2016 [12] and RuSentiment [20]. In this paper we focus only on the second dataset, its description is set out in **section 3.2**.

All the word representations before ELMo were context-independent. Although some of them take into account sub-word information [1] or learn sense-depended word vectors to solve lexical ambiguity problem, none of the approaches consider context for word representation. Announced in [17] high performance of embeddings from language models applied to most of NLP tasks, specifically text classification, textual entailment, named entity recognition, question answering, coreference resolution and semantic role labelling opened a new room for research. In recently published paper [9] authors achieve state-of-the-art results on named entity recognition built upon Russian ELMo.

ELMo's achievements induced popularity of transfer learning approach when complex architecture pre-trained on language modelling task should be fine-tuned for solution of some other supervised problem [18].

3. Data

3.1. Language modelling data

The Russian language models corresponding to official language style were trained on Wikipedia[1] and Russian WMT News[2] while the Russian conversational language model was trained on Russian twits[3]. Clue characteristics of the datasets are presented in Table 1.

Table 1: Data characteristics

Dataset	Number of words	Vocabulary size	Average number of words per sentence	File Size
Wiki	472 M	5.6 M	19.4	4.8 Gb
WMT News	1,133 M	4.1 M	19.6	12.0 Gb
Twitter	887 M	11.3 M	8.7	7.9 Gb

Preprocessed and cleaned WMT News sets are available for downloading, Wikipedia was spared from html-markup, and all hashtags and user logins were replaced by special tokens in Twitter. The vocabulary size for each dataset was set to 1 million frequency tokens. Finally, every dataset was splitted on training (98%) and validation (2%) samples.

3.2. Classification data

RuSentiment was published in 2018 [20] along with baseline results. The full dataset contains more than 30 thousands social media posts of average length 17 tokens, each post is related to one of five classes: positive, negative, neutral, speech and skip. Currently this is the largest publicly available dataset on Russian sentiment analysis. Around 21 thousands posts were randomly selected, and almost 7 thousands were pre-selected with an active learning-style strategy in order to diversify the data. We divide "random posts" subset on train and validation sets in a ratio of 9/1. The "pre-selected posts" set is not used in this paper. The test set is the same as in the original paper.

Linguists emit five Russian language styles: scientific, official, journalistic, artistic and colloquial. The first four styles and the last one differ a lot in terms of vocabulary and morphology. Therefore, we chose RuSentiment as the target dataset in this paper because the content relates to conversational style which often is not included to language modelling data while it is of current interest due to increasing popularity of chat-bots.

---

1 https://ru.wikipedia.org/

2 http://www.statmt.org/

3 https://twitter.com/

4. Experiments and Results

In this paper we explore the following token embeddings to cover different language styles:

- fastText embeddings trained on Russian Wiki and News corpora,
- fastText embeddings trained on Russian Twitter corpus,
- ELMo trained on Russian WMT News dataset,
- ELMo trained on Russian Wikipedia dataset,
- ELMo trained on Russian Twitter dataset,
- ELMo trained on Russian Twitter dataset and fine-tuned on RuSentiment.

300-dimensional fastText embeddings were trained with default parameters for skipgram model taking into account character n-grams from 3 to 6 characters.

4.1. Training and fine-tuning of language models

Language model consists of two main components: convolutional layers and 2 blocks of two recurrent layers. In the original implementation model receives as input indices of symbols in utf-8 encoding (from 0 to 255 plus three special symbols for padding, start and end of word). LSTM blocks pass forth and back over representations from convolutional layers, each block in its own direction similarly to bidirectional LSTM.

Training is being done in the similar to [6] and [8] way. An additional feed-forward layer followed by softmax is used to train language model. The model predicts words in direct and reverse orders for each LSTM blocks separately. The feed-forward layer is not used anymore after language model was fitted. To obtain context-dependent word representation weighted sum of word representations from all layers is used. Coefficients of this sum can be trained, and then can be different for all tasks. The upper layer also can be used similarly to TagLM [16] and CoVe [13]. Sentence representation is often formed as average or TF-IDF weighted sum [19] of word vectors.

This paper used model 4096/512 with 93.6 million of parameters[4]. The results of training language models on Wikipedia, WMT News, Twitter and fine-tuning of Twitter language model on RuSentiment data are presented in **Table 2**. Every language model was trained for 10 epochs in parallel on three 1080ti. Fine-tuning was conducted up to validation perplexity increase. The resulting perplexity of language model on "random posts" set of RuSentiment is 159.2 which was achieved after 4 epochs before overfitting began. The pre-trained language models were tested on full "random posts" set of RuSentiment. The resulting perplexity values are presented in **Table 2** in the last column. The language model trained on Twitter corpus performs best on RuSentiment dataset that was expected as language styles of corpora coincide.

[4] https://allennlp.org/elmo, https://s3-us-west-2.amazonaws.com/allennlp/models/elmo/2x4096_512_2048cnn_2xhighway/elmo_2x4096_512_2048cnn_2xhighway_options.json

Table 2: Results of training and fine-tuning ELMo

Data	Training time	Epochs	Perplexity on valid	Perplexity on RuSentiment
Wiki	6 days	10	43.692	17,364.89
WMT News	14 days	10	49.876	360.97
Twitter	10 days	10	94.145	172.25
Fine-tuning of Twitter on RuSentiment	15 min	4	159.2	—

Table 3 is presented for rough and fast estimation of the selected datasets similarity. As a metric of comparison, a perplexity of a bi-gram language model was chosen. The bi-gram model is to predict the conditional probability $P(w_n|w_{n-1})$ of a word w_n given the preceding word w_{n-1}. A KenLM [3] was used as an implementation of the fast N-gram language model. The resulting perplexity values of bi-gram models trained on a corresponding dataset are diagonal elements of Table 3. Other elements show how accurately a bi-gram model from one specific domain (rows) predicts words of test set from another specific domain (columns). As shown in Table 3 the Twitter bi-gram language model predicts words of RuSentiment significantly better than those trained on WMT News and Wiki. Simultaneously, RuSentiment bi-gram model predicts words of Twitter dataset with quality comparable to model trained on Twitter.

Table 3: The perplexity of word bi-gram models on testing sets

Bi-gram model\Data	RuSentiment	WMT News	Twitter	Wiki
RuSentiment	116.67	4,847.68	**9,094.83**	**7,151.52**
WMT News	369,864.24	640.55	434,928.31	10,381.87
Twitter	**46,657.95**	1,740.06	6,762.07	8,330.85
Wiki	189,929.95	**1,583.86**	197,762.66	1,586.13

4.2. Training classifiers

There are two main approaches for text classification: convolutional and recurrent networks. Therefore, consider SWCNN [7] and BiGRU [2], [5] basic architectures of this paper.

The first model, shallow-and-wide convolutional neural network (SWCNN) illustrated in Fig. 1, sends non-trainable token embeddings to three convolutions with the same number of filters and different kernel sizes, each of which is followed by batch normalization layer [4], ReLU activation and global max pooling to reduce dimensionality. Pooled outputs are concatenated along the last dimension, and given to dense layer followed by batch normalization and ReLU activation. The output is given to classification dense layer also followed by batch normalization and softmax activation. Two dropout layers are placed directly before dense layers, and kernels are L2-regularized [14].

Bidirectional GRU (BiGRU) is demonstrated in Fig. 2. Non-trainable token embeddings are sent to bidirectional GRU layer which is followed by global max and average pooling. Pooled outputs are concatenated with two last states from BiGRU, and sent to dense layer followed by ReLU activation. Then output is given to the last classification dense layer followed by softmax activation. Two dropout layers are placed directly before dense layers, and kernels are also L2-regularized.

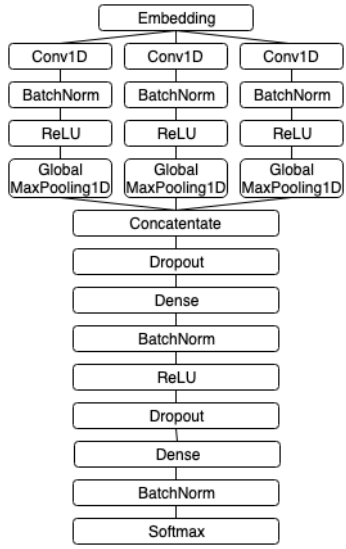

Figure 1: Shallow-and-wide CNN

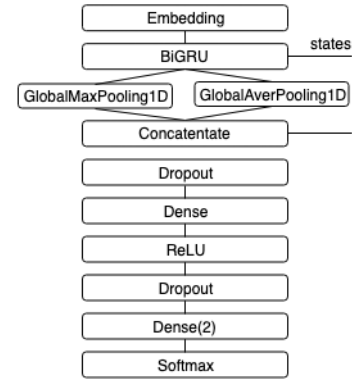

Figure 2: Bidirectional GRU

Baseline models are two networks of the above described architectures trained upon pre-trained fastText embeddings of dimensionality 300. The fastText skipgram model of official language style was trained on Russian Wikipedia and news corpora, fastText skpigram conversational style model was trained on Twitter dataset, both fastText models are available for downloading[5]. To explore domain-dependency of language models we also consider neural networks receiving token ELMo representations of dimensionality 1,024. The target metric is weighted F1-score, training is due to excess of patience limit.

All the experiments were conducted with the same parameters. Convolutional layers had 256 filters and kernels of sizes 3, 5, 7 while BiGRU layer had 256 units. The first dense layer had 100 units for both networks. Patience limit was set to 2, and maximum number of epochs was equal to 10. SWCNN models were strongly regularized with dropout rate of 0.5 and L2-coefficients 10^{-3} and 10^{-2} for convolutional and dense kernels. BiGRU model had dropout rate of 0.2, and L2-coefficient 10^{-6} for both recurrent and dense kernels.

[5] http://docs.deeppavlov.ai/en/latest/intro/pretrained_vectors.html

Table 4: Resulting scores on RuSentiment with different embeddings

Model	Embeddings	Validation F1-weighted	Test F1-weighted
Rogers et al. [20]	fastText VK	—	72.80
SWCNN	fastText Wiki+News	67.84	70.27
BiGRU	fastText Wiki+News	69.54	71.74
SWCNN	fastText Twitter	70.91	73.03
BiGRU	fastText Twitter	72.62	74.45
SWCNN	ELMo WMT News	70.27	72.42
BiGRU	ELMo WMT News	70.15	71.37
SWCNN	ELMo Wiki	68.11	71.28
BiGRU	ELMo Wiki	66.55	69.47
SWCNN	ELMo Twitter	75.40	78.50
BiGRU	ELMo Twitter	75.89	77.62
SWCNN	ELMo Fine-tuned	74.74	77.98
BiGRU	ELMo Fine-tuned	75.75	77.19

Each experiment was run for 4 times, the resulting averaged weighted F1-scores are presented in Table 4. For fastText embeddings BiGRU shows better than SWCNN results while for ELMo convolutional models outperform recurrent. Embedding models corresponding to official and journalistic language styles have almost the same scores with original paper [20] (71.7 weighted F1-scores when "pre-selected posts" were not used). Although fastText embeddings trained on Twitter dataset for both architectures beat not only baseline from [20] but all the models trained on domains of official (Wiki) and journalistic (News) styles, they are significantly transcended by conversational (Twitter) embeddings from language models. The best results (almost 6 points higher than previous state-of-the-art) are enriched by shallow-and-wide convolutional network trained on top of embeddings from Twitter language model.

5. Discussion

We have trained two popular architectures on 6 different embeddings of official, journalistic and conversational language styles. As the domain of target sentiment classification dataset is related to conversational language it was expected to obtain better results for conversational embeddings but the rate of the increase of scores is dramatic. Embeddings from language models not only appropriate but obligatory to be used in classification tasks if the domain of language model and target problem are close. Let us demonstrate several examples which support the statement in Table 5. One can pay attention to lexicon of the presented test samples, and which domain of language embeddings is closer than others.

Table 5: Examples of mistakes of models trained on top of different embeddings

Text sample	True label	ELMo News	ELMo Wiki	ELMo Twitter
василий зе бест!	*positive*	skip	skip	*positive*
вкусняшка, омном-ном	*positive*	neutral	skip	*positive*
полнейший зашквар назначать некогда хорошего футболиста сразу главным тренером «реала»	*negative*	neutral	neutral	*negative*
я променяла вас на диплом! а еще на министерское тестирование и гос экзамены!!я 0 числа уже с дипломом в зубах буду!!	*positive*	*positive*	skip	*negative*
все! завтра улетаю на евро- 0 в польшу болеть за сборную россии!	*positive*	*positive*	neutral	neutral
ну кто еще теперь задаст вопросы «зачем нами эта олимпиада?» «зачем нам спорт высоких достижений?». ведь можем же, когда захотим...	neutral	*negative*	neutral	*negative*

To summarize, we have introduced pre-trained Russian language models which allow to perform better, and to be evidential we have demonstrated how embeddings from language model outperform common fastText embeddings in Russian sentiment analysis task. Simultaneously, we have shown how significant the dependency of quality on the language model's domain is.

Acknowledgements

This work was supported by National Technology Initiative, and PAO Sberbank project ID 0000000007417F630002.

References

1. *Bojanowski, P. et al.:* Enriching word vectors with subword information. Transactions of the Association for Computational Linguistics. 5, 135–146 (2017).
2. *Cho, K. et al.:* Learning phrase representations using rnn encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078. (2014).
3. *Heafield, K. et al.:* Scalable modified Kneser-Ney language model estimation. In: Proceedings of the 51st annual meeting of the association for computational linguistics. pp. 690–696, Sofia, Bulgaria (2013).
4. *Ioffe, S., Szegedy, C.:* Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: International conference on machine learning. pp. 448–456 (2015).

5. *Johnson, R., Zhang, T.:* Supervised and semi-supervised text categorization using lstm for region embeddings. arXiv preprint arXiv:1602.02373. (2016).
6. *Jozefowicz, R. et al.:* Exploring the limits of language modeling. arXiv preprint arXiv:1602.02410. (2016).
7. *Kim, Y.:* Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882. (2014).
8. *Kim, Y. et al.:* Character-aware neural language models. In: Thirtieth aaai conference on artificial intelligence. (2016).
9. *Konoplich, G. et al.:* Named entity recognition in russian with word representation learned by a bidirectional language model. In: Conference on artificial intelligence and natural language. pp. 48–58 Springer (2018).
10. *Loukachevitch, N. et al.:* SentiRuEval: Testing object-oriented sentiment analysis systems in russian. In: Proceedings of international conference dialog. pp. 3–13 (2015).
11. *Loukachevitch, N. V., Levchik, A.:* Creating a general russian sentiment lexicon. In: LREC. (2016).
12. *Lukashevich, N., Rubtsova, Y. V.:* SentiRuEval-2016: Overcoming time gap and data sparsity in tweet sentiment analysis. In: Компьютерная лингвистика и интеллектуальные технологии. pp. 416–426 (2016).
13. *McCann, B. et al.:* Learned in translation: Contextualized word vectors. In: Advances in neural information processing systems. pp. 6294–6305 (2017).
14. *Ng, A. Y.:* Feature selection, l 1 vs. l 2 regularization, and rotational invariance. In: Proceedings of the twenty-first international conference on machine learning. p. 78 ACM (2004).
15. *Pennington, J. et al.:* Glove: Global vectors for word representation. In: Proceedings of the 2014 conference on empirical methods in natural language processing (emnlp). pp. 1532–1543 (2014).
16. *Peters, M. E. et al.:* Semi-supervised sequence tagging with bidirectional language models. arXiv preprint arXiv:1705.00108. (2017).
17. *Peters, M. E. et al.:* Deep contextualized word representations. arXiv preprint arXiv:1802.05365. (2018).
18. *Radford, A. et al.:* Improving language understanding by generative pre-training. URL https://s3-us-west-2. amazonaws. com/openai-assets/research-covers/languageunsupervised/language understanding paper. pdf. (2018).
19. *Robertson, S.:* Understanding inverse document frequency: On theoretical arguments for idf. Journal of documentation. 60, 5, 503–520 (2004).
20. *Rogers, A. et al.:* RuSentiment: An enriched sentiment analysis dataset for social media in russian. In: Proceedings of the 27th international conference on computational linguistics. pp. 755–763 (2018).