USE OF MORPHOLOGY IN DISTRIBUTIONAL WORD EMBEDDING MODELS: RUSSIAN LANGUAGE CASE

Sadov M. A. (mikeabyrvalg5@gmail.com) – NRU HSE, Moscow, Russia; Kutuzov A. B. (andreku@ifi.uio.no) – University of Oslo, Oslo, Norway.

Most of the distributional word embedding models nowadays learn semantic representations of words ignoring their morphological structure. This became a limitation, especially for languages with complex morphology: their vocabularies are quite large, and most words are infrequent which results in models being unable to learn good semantic representations for such words. In this paper, we compare two approaches aimed at including subword information for Russian using distributional word embedding models trained on the Russian National Corpus. We evaluate these approaches on the newly created rare and multimorphemic word similarity dataset, which itself is another contribution of ours. Overall, we show the benefits of using subword informations for learning better semantic representations of words.

Key words: distributional semantics, word embeddings, word2vec, fastText, semantic similarity, subword distributional models, evaluation sets.

ИСПОЛЬЗОВАНИЕ МОРФОЛОГИИ В ДИСТРИБУТИВНЫХ СЕМАНТИЧЕСКИХ МОДЕЛЯХ: ЭКСПЕРИМЕНТЫ С РУССКИМ ЯЗЫКОМ

Садов М. А. (mikeabyrvalg5@gmail.com) – НИУ ВШЭ, Москва, Россия; Кутузов А. Б. (andreku@ifi.uio.no) – Университет Осло, Осло, Норвегия.

1. Introduction

The first attempts to derive representations of word meaning were based on directly extracting co-occurrence statistics from large corpora [Deerwester et al., 1990]: each word is represented with a vector that consists of frequencies for this word occurring together with other words. The assumption is that such vectors for semantically similar words are close to each other (by the cosine similarity metric) because the words are used in similar contexts. This group of methods is called *count-based* since they calculate the co-occurrences of each word in the corpus with all other words directly. However, it is *prediction-based* methods that have attracted most attention in the field of distributional semantics in the recent years: they approximate co-occurrence statistics without counting it directly, using machine learning, particularly shallow neural networks. Two well-known state-of-the-art approaches in this group are *Continuous Bag-of-Words* (CBOW) and *SkipGram* introduced in [Mikolov et al., 2013a].

Currently most of the distributional word embedding models (“embedding” stands for a mathematical structure that is used to embed word meaning into a dense vector)

learn semantic representations of words ignoring their morphological structure. This became a limitation, especially for languages with complex morphology: considering the facts that their vocabularies are large and most words are infrequent, it is quite probable that a model trained for the one of such languages either will learn distorted semantic representations for these words or will not be able to learn them at all [Bojanowski et al., 2017].

In this paper, we compare two approaches towards including subword information for Russian using distributional word embedding models trained on the Russian National Corpus[1]: the first approach is based on two models trained using *Continuous SkipGram* algorithm (the main model does not employ any subword information and the additional model extends the main one by representing meanings of out-of-vocabulary words through combining representations of morphemes of such words) and the second approach that uses a model trained with *fastText* algorithm [Bojanowski et al., 2017]. We evaluate the approaches on a rare and multimorphemic word similarity dataset, which we compiled. This is another contribution of our work.

The paper is structured as follows. In Section 2 we put our research in the context of the previous work. Section 3 introduces the corpora, the methods used to train distributional word embedding models and the models to segment words in morphs. In Section 4 we describe the process of creation of our word similarity dataset. Section 5 presents the evaluation results and the analysis of mistakes, and in Section 6 we conclude.

2. Related work

In recent years, the works that proposed methods for the usage of subword information in distributional word embedding models (further DWEM or DWEMs) were based mostly on the English [Luong et al., 2013; Xu et al., 2017; Bojanowski et al., 2017] and German [Padó et al., 2016; Singh et al., 2016] material.

One of the first attempts to employ subword information was made in [Luong et al., 2013; Padó et al., 2016]: the key idea was to represent the meaning of an out-of-vocabulary (further OOV) word by combining semantic vector representations of its morphemes. In order to segment words into morphemes, the researchers used either manually written rules or the *Morfessor* tool [Smit et al., 2014]. [Luong et al., 2013] claimed that *Morfessor* is especially handy for words in morphologically rich languages such as Finnish and Turkish so it is quite possible that this tool can be also applicable to the analysis of Russian. The other way to retrieve morphological segments was described in [Ruokolainen et al., 2014]: the authors used supervised and semi-supervised linear-chain CRF-based algorithms to segment words. The idea

[1] [1] http://www.ruscorpora.ru/en

to use morphemes was further elaborated in [Xu et al., 2017] and [Filchenkov, 2017], where the authors tried to implicitly use morphological information by linking each morpheme with the word that defines the semantic meaning of this morpheme (for example, representing the affix “micro” with the word “small”).

Another way to solve the problem of modeling OOV words meaning was proposed in [Singh et al., 2016]. For each unknown word, the authors searched for the words from the model dictionary that have as many common character 3-grams with this target word as possible. After that, the meaning of the unknown word was represented as the weighted average of top 10 words found during the search. [Bojanowski et al., 2017], in turn, introduced *fastText* algorithm that is based on the idea of using character n-grams combinations of different length as subword information modifying the existing *SkipGram* architecture. In this paper, the authors used the materials of several languages including English, German, Spanish, French and Czech, and claimed to achieve state-of-the-art performance in the tasks of word similarity and analogy, which makes their approach particularly interesting to compare with the ones that use morphemes to represent the meanings of OOV words.

3. Resources used

3.1 Corpora

To train all the models (except for the supervised CRF-based morpho-segmentation model which was trained on the part of Tikhonov’s morphological dictionary [Tikhonov, 1990]) and to compile the word similarity dataset, we used the Russian National Corpus (further RNC), which is the flagship academic corpus of Russian. The data from the corpus was lemmatized with *Mystem* [Segalovich, 2003], all punctuation was removed and all words were lower-cased. An additional filtering procedure we employed was deletion of all the sentences which contained less than three words.

The final size of the training corpus for the additional DWEM trained on morphemes from the CRF-based model is 124 961 390 word tokens and 10 099 999 sentences, for all other DWEMs it is 161 044 270 word tokens and 12 994 398 sentences.

3.2 Training algorithms

The DWEMs that employ subword information in the form of morphemes were trained using Word2Vec (a widely used instrument for training DWEMs) *Continuous SkipGram* algorithm with vector size 300, Negative Sampling approach [Mikolov et al., 2013b] with the number of negative samples set to 15 and a symmetric context window of 5 words to the left and 5 words to the right. Lemmas served as input for the main model that does not include any subword information while the additional

models were fed in the following way: one – with morphs received after segmenting lemmas by *Morfessor*, the other – with morphs from CRF-based morpho-segmentation model.

The choice of the vector size was motivated by the fact that generally this size is perceived as optimal for semantic vectors [Jurafsky, 2000]; for more information on how the vector size influences the performance of models we refer the reader to [Kutuzov and Andreev, 2015].

Speaking about the number of negative samples, [Mikolov et al., 2013b] stated that the best performance is achieved when the value of the parameter lies in the range from 5 to 20 if a model is trained on a big amount of data and that was proved by the results of the tests conducted by the author: the highest performance was shown by the model that was trained on 1 billion word tokens with 15 negative samples. This conclusion is also supported by [Levy et al., 2015]: while using Negative Sampling approach it is preferable to use many negative samples.

The reason for choosing relatively wide window size was the following: despite the known fact that larger windows induce the models that are more “associative” [Levy and Goldberg, 2014], the models which use narrow windows can miss important parts of context for some words, especially in the case of discontinuous constituents. Moreover, the words that are situated far from the focus one will have little impact on the final representations compared to the closest words and the actual window for a fair amount of words will be less than 5 words from the both sides. There are several reasons for that: use of Word2Vec which employs weighting scheme by the distance from the focus word divided by the window size [Levy et al., 2015] and limitations put on the number of context words induced by the choice of sentence as a context item. Considering all these facts, the models will be less “associative” than it can be expected.

All the tokens that appeared in the training corpus less than 3 times were discarded. Down-sampling parameter was set to the default value (equal to 0.001) because there was no stop-words removal procedure at the stage of data preparation. The use of this parameter can widen the context window because the removal of frequent word tokens in Word2Vec is done before the corpus is processed into word-context pairs (so-called “dirty” down-sampling). However, [Levy et al., 2015] stated that the impact of “dirty” down-sampling on the performance is comparable to the “clean” down-sampling and hence cannot dramatically influence the results. During the training, the algorithm iterated over the corpus 5 times. In the end of the training, the main model that does not include any subword information contained vectors for 276 463 words, whereas the additional model with Morfessor had 11 114 vocabulary units and the one with CRF – 138 952.

The parameters used to train the *fastText* model that uses subword information in the form of character n-grams were almost the same as for the two previous ones except for the frequency threshold: it was set to 5. It was done to reduce the size of the model. The expected information losses induced by this decision are minimal, because the information that the *fastText* model learned from all character n-grams of words should in theory be able to compensate for the losses. In the end of the training, the *fastText* model contained vectors for 202 396 words.

We used n-grams of length 4 and 5. Clearly, this parameter can vary depending on the task and the language, however the investigation of this dependency lies out of the scope of this work. The key parameters used for training our models are shown in Table 1.

Model name	**Vector size**	**Context window**	**Minimal count**	**Negative samples**	**N-grams range**	**# of vectors**
main model	300	10	3	15	~	276 463 (lemmas)
additional model (Morfessor)	300	10	3	15	~	11 114 (morphemes)
Additional model (CRF-based)	300	10	3	15	~	138 952 (morphemes)

fastText	300	10	5	15	4-5	202 396 (lemmas), 404245 (ngrams)

Table 1. The key parameters of the models

3.3 The morpho-segmentation algorithms

The first model for segmenting words in morphemes was trained using the *Morfessor* tool. The method chosen for training the model is Morfessor Baseline [Creutz and Lagus, 2002; Creutz and Lagus, 2005], which is a context-independent splitting algorithm. The model was trained in batch fashion with the following parameters: recursive algorithm, no words discarded and no count modifier function for adjusting the counts of words.

The second model for morpho-segmentation is supervised morphological linear-chain CRFs model described in [Ruokolainen et al., 2014]. We used Tikhonov's morphological dictionary as training data for this model: we randomly picked 828 entries as train set and 516 – as development set.

4. Gold standard word similarity dataset

4.1 Word pairs formation

The selection of rare and multimorphemic words was executed in two steps: first, the candidates were filtered by the number of morphemes segmented by *Morfessor* and, second, according to the word frequency.

A candidate word was considered as multimorphemic if it was segmented into 5 or more morphemes. Only the candidate words that occurred in the RNC less than 1 000 times were picked. [Luong et al., 2013] set the maximum frequency threshold for the formation of rare word dataset to 10 000, but they used the English Wikipedia, which is obviously several times larger than the RNC. The frequency data was collected through querying the RNC website search engine.

After the selection procedure, the words were divided into three groups according to their frequency: *extremely rare* [1, 10], *moderately rare* [11, 100], and *common* [101, 1 000]. Then 40 words were randomly picked from each group, but during the manual control procedure it was discovered that some of the words are not suitable for the gold standard because it would be difficult for the human experts to define their similarity to another word in pair. This category of words consisted of

possessive adjectives that denote belonging to some entity (for example, "*мефистофелевский*" ("mephistophelic") and "*стерлитамакский*" ("belonging to Sterlitamak")) and complicated names of chemical compounds. All the words of this category belonging to the group of extremely rare were removed and the ones that belonged to the group of moderately rare were replaced in the final list by the words belonging to the common group. After that, the total number of words was 104, from which 46 are common, 34 are moderately rare, and 24 are extremely rare. The distribution of parts of speech was the following: 73 adjectives (participles also fall in this category), 23 nouns, 6 verbs and 2 adverbs.

For word pairs formation, we planned to use *RuThes Lite* [Panchenko et al., 2016] lexical database. However, only 3 words from our list of rare and multimorphemic words were found in *RuThes Lite* and hence for all the remaining 101 words the pairs were formed manually. The main criteria for word pairs formation was the same part-of-speech tag for both words in a pair because as it was shown in [Markman and Wisniewski, 1997], when comparing words in noun – noun pairs and verb – verb pairs, different cognitive operations are employed. It means that the presence of the pairs with different part-of-speech tags can make the annotation task more complicated for the human experts.

4.2 Annotation procedure

To collect the semantic similarity scores, crowdsourcing with the Google Forms web service was used. The survey was divided in 3 parts: in the first part the participants had to read the annotators guidelines[2], in the second part they were asked to submit their personal data (age, sex and level of education) and in the third part they started to annotate. The pairs were grouped into 18 groups and each of the groups contained 5 or 6 pairs. The final version of the word similarity dataset is organized in the following way: 104 lines, where each line consists of a word pair and 13 scores of its semantic similarity proposed by the annotators.

4.3 Inter-annotator agreement measure

In order to measure the inter-annotator agreement, the Krippendorff's alpha [Krippendorff, 2011] metric was used: it was computed for each pair of annotators and then the overall mean was calculated. The score was 0.648 which is high enough to state that our dataset is a reliable evaluation sample for measuring performance of DWEMs on rare and multimorphemic words.

5. Results

[2] The full guidelines are available at https://github.com/sadov-m/DSM_morphology/blob/master/golden_standard/annotation_guide.docx

For measuring the performance of the models, Spearman's rank correlation coefficient [Spearman, 1904] was used. We chose this method of evaluation instead of precision and recall metrics, because they are not suitable for semantic similarity task: DWEMs performance is as a rule estimated by calculating the rank correlation between pairwise scores from the model and from human judgements [Hill et al., 2015].

5.1 Performance with Morfessor

The proportion of pairs that contain OOV words for the *SkipGram+Morfessor* model is approximately 17.3 percent (18 words in 18 pairs) whereas for the *fastText* model it is 19.2 percent (20 words in 20 pairs).

Model name	P	p-value	OOV words	total words
SkipGram+Morfessor	0.5369	4.2125e-09	17.3%	104
fastText	0.7337	8.0981e-19	19.2%	104

Table 2.1: Spearman's rank correlation values (ρ) between the human experts scores and models estimations on the full word similarity dataset (Morfessor experimental set-up).

As can be seen in Table 2.1, the *fastText* model slightly outperformed the *SkipGram+Morfessor* model despite having more OOV words to predict and the results are reliable at the 99.9 percent confidence level. After the tests on the full sample the sets of OOV words belonging to the word similarity dataset for both models were intersected, and the test was repeated on OOV words only. The results are shown in Table 2.2 and now the leadership of the *fastText* model is much more obvious. Despite little amount of words in this testset it is obvious that *SkipGram+Morfessor* model was not able to model the meanings of OOV words at all. The main reason for that can be found in the segmentations made by *Morfessor*.

Model name	P	p-value	OOV words	total words
SkipGram+Morfessor	-0.1647	0.5134	100%	18
fastText	0.7176	0.0007	100%	18

Table 3: Spearman's rank correlation values (ρ) between the human experts scores and models estimations on the common set of OOV words (Morfessor experimental set-up).

The Morfessor Baseline algorithm used in this work has several limitations. The most important one is its contextual independence which resulted in our Morfessor model

violating morpho-tactic rules: for example, it suggested the prefix "по" in non-word-initial positions ("вал/ян/оса/по/ж/ник", "felt boots maker"). The other limitation came from the fusion nature of Russian: our Morfessor model performed well on the words with complex but consecutive morphology (for example, "*полу/рас/па/вший/ся*", "half-disintegrated"), but predictably made mistakes in the cases with fusion ("*сте/речь*", "to guard"). The last typical error of our Morfessor model was over-segmentation of strings which include frequent morphemes ("*ра/б*", "slave").

5.2 Performance with CRF-based morphological model

Table 3.1 shows almost the same figures: the *fastText* model results was again superior to the ones of *SkipGram+CRF-based* model.

Model name	P	p-value	OOV words	total words
SkipGram+CRF-based	0.5344	5.0973e-09	17.3%	104
fastText	0.7337	8.0981e-19	19.2%	104

Table 3.1: Spearman's rank correlation values (ρ) between the human experts scores and models estimations on the full word similarity dataset (CRF-based model experimental set-up).

However, the results of the test on OOV words only in Table 3.2 show that *SkipGram+CRF-based* model performed even worse despite the higher quality of morphological segmentation provided by CRF-based model.

Model name	P	p-value	OOV words	total words
SkipGram+CRF-based	-0.3244	0.189	100%	18
fastText	0.7176	0.0007	100%	18

Table 3.2: Spearman's rank correlation values (ρ) between the human experts scores and models estimations on the common set of OOV words (CRF-based model experimental set-up).

6. Conclusion

We compared two approaches of including subword information into Russian word embedding models:

1. the morphological approach including two *SkipGram* DWEMs;
2. the character n-grams approach based on the *fastText* DWEM.

All the models were trained on the RNC. Additionally, we introduced the word similarity dataset of rare and multimorphemic words for Russian and evaluated our models against it. We publish all the models[3] [4] [5] [6] together with the gold standard dataset[7].

After the evaluation, we discovered that the *fastText* model showed great results despite having more OOV words to predict. On the contrary, both SkipGram models failed at learning representations of OOV words.

Our word similarity dataset features a good inter-annotator agreement score (α = 0.648) which permits us to state that it is suitable for measuring the performance of DWEMs on rare and multimorphemic words.

We also revealed several flaws in the Morfessor Baseline algorithm and came to the conclusion that the tool performance on complex words with fusion is relatively poor and the algorithm itself has too many limitations to be considered reliable.

As a future work, we plan to study in more detail how different morpho-segmentation approaches affect the performance of the models trained on morphemes. We also would like to widen our word similarity dataset of rare and multimorphemic words for Russian.

References

[Bojanowski et al., 2017] Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the Association of Computational Linguistics*, 5:135–146.

[Creutz and Lagus, 2002] Creutz, M., & Lagus, K. (2002, July). Unsupervised discovery of morphemes. In *Proceedings of the ACL-02 workshop on Morphological and phonological learning-Volume 6* (pp. 21-30). Association for Computational Linguistics.

[Creutz and Lagus, 2005] Creutz, M., & Lagus, K. (2005). *Unsupervised morpheme segmentation and morphology induction from text corpora using Morfessor 1.0*. Helsinki: Helsinki University of Technology.

[3] https://drive.google.com/file/d/0B60o6LzcMfuBMVQxRDNRR0UxZXM
[4] https://drive.google.com/file/d/0B60o6LzcMfuBeDVnaFZLVHJzVTQ
[5] https://drive.google.com/open?id=1qXEMOS-LmGNaZ46UVlBxCltU0-YzulRI
[6] https://drive.google.com/open?id=1bx2zx6ljShKC6ncMM-9KxMSTISgO_VPB
[7] https://github.com/sadov-m/DSM_morphology/blob/master/morfessor/golden_standard.csv

[Deerwester et al., 1990] Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., & Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American society for information science, 41*(6), 391.

[Filchenkov, 2017] Filchenkov, A. (2017, November). Morpheme Level Word Embedding. In *Artificial Intelligence and Natural Language: 6th Conference, AINL 2017, St. Petersburg, Russia, September 20–23, 2017, Revised Selected Papers* (Vol. 789, p. 143). Springer.

[Jurafsky, 2000] Jurafsky, D. (2000). *Speech & language processing*. Pearson Education India.

[Hill et al., 2015] Hill, F., Reichart, R., & Korhonen, A. (2015). Simlex-999: Evaluating semantic models with (genuine) similarity estimation. Computational Linguistics, 41(4), 665-695.

[Krippendorff, 2011] Krippendorff, K. (2011). Computing Krippendorff's alpha-reliability.

[Kutuzov and Andreev, 2015] Kutuzov, A., & Andreev, I. (2015). Texts in, meaning out: neural language models in semantic similarity task for Russian. *arXiv preprint arXiv:1504.08183*.

[Levy and Goldberg, 2014] Levy, O., & Goldberg, Y. (2014). Dependency-based word embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)* (Vol. 2, pp. 302-308).

[Levy et al., 2015] Levy, O., Goldberg, Y., & Dagan, I. (2015). Improving distributional similarity with lessons learned from word embeddings. *Transactions of the Association for Computational Linguistics, 3*, 211-225.

[Luong et al., 2013] Luong, T., Socher, R., & Manning, C. (2013). Better word representations with recursive neural networks for morphology. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning* (pp. 104-113).

[Markman and Wisniewski, 1997] Markman, A. B., & Wisniewski, E. J. (1997). Similar and different: The differentiation of basic-level categories. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 23*(1), 54.

[Mikolov et al., 2013a] Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

[Mikolov et al., 2013b] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems* (pp. 3111-3119).

[Padó et al., 2016] Padó, S., Herbelot, A., Kisselew, M., & Šnajder, J. (2016). Predictability of distributional semantics in derivational word formation. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers* (pp. 1285-1296).

[Panchenko et al., 2016] Panchenko, A., Ustalov, D., Arefyev, N., Paperno, D., Konstantinova, N., Loukachevitch, N., & Biemann, C. (2016, April). Human and machine judgements for russian semantic relatedness. In *International Conference on Analysis of Images, Social Networks and Texts* (pp. 221-235). Springer, Cham.

[Ruokolainen et al., 2014] Ruokolainen, T., Kohonen, O., & Virpioja, S. (2014). Painless semi-supervised morphological segmentation using conditional random fields. In Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, volume 2: Short Papers (pp. 84-89).

[Segalovich, 2003] Segalovich, I. (2003, June). A fast morphological algorithm with unknown word guessing induced by a dictionary for a web search engine. In *MLMTA* (pp. 273-280).

[Singh et al., 2016] Singh, M., Greenberg, C., Oualil, Y., & Klakow, D. (2016). Sub-Word Similarity based Search for Embeddings: Inducing Rare-Word Embeddings for Word Similarity Tasks and Language Modelling. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers* (pp. 2061-2070).

[Smit et al., 2014] Smit, P., Virpioja, S., Grönroos, S. A., & Kurimo, M. (2014). Morfessor 2.0: Toolkit for statistical morphological segmentation. In *The 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL), Gothenburg, Sweden, April 26-30, 2014*. Aalto University.

[Spearman, 1904] Spearman, C. (1904). The proof and measurement of association between two things. *The American journal of psychology, 15*(1), 72-101.

[Tikhonov, 1990] Tikhonov, A. N. (1990). Word-Formation Dictionary of Russian: In 2 V. M.: The Russian Language.

[Xu et al., 2017] Xu, Y., & Liu, J. (2017). Implicitly incorporating morphological information into word embedding. arXiv preprint arXiv:1701.02481.