Enriching word embeddings with global information and testing on highly inflected language

Lukáš Svoboda[1 2] and Tomáš Brychcín[1 2]

[1] Department of Computer Science and Engineering, Faculty of Applied Sciences, University of West Bohemia, Univerzitní 8, 306 14 Plzeň, Czech Republic
[2] NTIS—New Technologies for the Information Society, Faculty of Applied Sciences, University of West Bohemia, Univerzitní 8, 306 14 Plzeň, Czech Republic
{svobikl,brychcin}@kiv.zcu.cz
nlp.kiv.zcu.cz

Abstract. In this paper we evaluate our new approach based on the *Continuous Bag-of-Words* and *Skip-gram* models enriched with global context information on highly inflected Czech language and compare it with English results. As a source of information we use Wikipedia, where articles are organized in a hierarchy of categories. These categories provide useful topical information about each article.

Both models are evaluated on standard word similarity and word analogy datasets. Proposed models outperform other word representation methods when similar size of training data is used. Model provide similar performance especially with methods trained on much larger datasets.

1 Introduction

The principle known as *Distributional hypothesis* is derived from the semantic theory of language usage, the meaning of words that are used and occur in the same contexts tend to have similar meaning [7]. The claim has the theoretical basics in psychology, linguistics, or lexicography [4]. This research area is often referred to as *distributional semantics*. During last years it has become a popular. Models based on this assumption are denoted as *distributional semantic models* (DSMs).

1.1 Distributional semantic models

DSMs learn contextual patterns from huge amount of textual data. They typically represent the meaning as a vector which reflects the contextual (distributional) information across the texts [34]. The words are associated with a vector of real numbers. Represented geometrically, the meaning is a point in a k-dimensional space. The words that are closely related in meaning tend to be closer in the space. This architecture is sometimes referred to as the *semantic space*. The vector representation allows us to measure similarity between the meanings (most often by the cosine of the angle between the corresponding vectors).

Word-based semantic spaces provide impressive performance in a variety of NLP tasks, such as language modeling [2], named entity recognition [14], sentiment analysis [11], and many others.

1.2 Local versus global context

Different types of context induce different kinds of semantic spaces. [26] and [20] distinguish *context-word* and *context-region* pproaches to the meaning extraction. In this paper we use the notation *local context* and *global context*, respectively. Global-context *DSMs* are usually based on *bag-of-words hypothesis*, assuming that the words are semantically similar if they occur in similar articles and the order in which they occur in articles has no meaning. These models are able to register long-range dependencies among words and are more topically oriented. In contrast, local-context *DSMs* collect short contexts around the word using moving window to induce the meaning. Resulting word representations are usually less topical and exhibit more functional similarity (they are often more syntactically oriented).

To create a proper *DSM* a large textual corpus is usually required. Very often Wikipedia is used for training, because it is currently the largest knowledge repository on the Web and is available in dozens of languages. The most of current *DSMs* learn the meaning representation merely from the word distributions and does not incorporate any metadata which Wikipedia offer.

1.3 Our model

In this work we combine both the local and the global context to improve the word meaning representation. We use local-context *DSMs Continuous Bag-of-Words* (CBOW) and Skip-Gram models [21], the original tool is often denoted as *Word2Vec*. We incorporate Wikipedia categories as a global context.

We train our models on English and Czech Wikipedia. We evaluate it on standard word similarity and word analogy datasets. Proposed models significantly outperform other word representation methods when similar training data size is used and provide similar performance compared with methods trained on much larger datasets.

1.4 Outline

The structure of article is following. Section 2 puts our work into the context of the state of the art. In Section 3 we review Word2Vec models on which our work is based. We define our model in Section 5 and 4. The experimental results presented in Section 7. We conclude in Section 8 and offer some directions for future work.

2 Related Work

In the past decades, simple frequency-based methods for deriving word meaning from raw text were popular, e.g. Hyperspace Analogue to Language [18] or Correlated Occurrence Analogue to Lexical Semantics [27] as a representatives of local-context DSMs and Latent Semantic Analysis [16] or Explicit Semantic Analysis [8] as a representatives of global-context DSMs. All these methods record word/context co-occurrence statistics into the one large matrix defining the semantic space.

Later on, these approaches have evolved in more sophisticated models. [21] revealed neural network based model *CBOW Skip-gram* that we are going to use as our baseline to incorporate Global context. His simple single-layer architecture is based on the inner product between two word vectors (detailed description is in Section 3). [25] introduced Global Vectors, the log-bilinear model that uses weighted least squares regression for estimating word vectors. The main concept of this model is the observation that global ratios of word/word co-occurrence probabilities have the potential for encoding meaning of words.

2.1 Local context with subword information

Above mentioned models currently serve as a basis for many researches. [1] improved Skip-Gram model by incorporating a sub-word information. Simirarly, in most recent study [30] incorporated a sub-word information into LexVec [29] vectors. This improvement is especially evident for languages with rich morphology. [17] used syntactic contexts automatically produced by dependency parse-trees to derive the word meaning. Their word representations are less topical and exhibit more functional similarity (they are more syntactically oriented).

[13] presented a new neural network architecture which learns word embeddings that capture the semantics of words by incorporating both local and global document context. It accounts for homonymy and polysemy by learning multiple embeddings per word. Authors introduce a new dataset with human judgments on pairs of words in sentential context, and evaluate their model on it. Their approach is focusing on polysemous words and generally do not perform as well as Skip-Gram or CBOW.

3 Word2Vec

This section describes Word2Vec package which utilizes two neural network model architectures (CBOW and Skip-Gram) to produce a distributional representation of words [21]. Given the training corpus represented as a set of documents $\boldsymbol{D}$. Each document (resp. article) $\boldsymbol{a}_j \in \boldsymbol{D}$ is a sequence of words $\boldsymbol{a}_j = \{w_{j,i}\}_{i=1}^{L_j}$, where L_j denote the length of the article $\boldsymbol{a}_j$. Each word w in the vocabulary $\boldsymbol{W}$ is represented by two different vectors $\mathbf{v}$ and $\mathbf{u}$ depending whether it is used as a context word $\mathbf{v}_w \in \mathbb{R}^d$ or a target word $\mathbf{u}_w \in \mathbb{R}^d$. The task is to estimate these vector representations in a way that optimize bellow described objective functions.

We use training procedure introduced in [22] called *negative sampling*. For the word at position i in the article $\boldsymbol{a}_j$ we define the negative log-likelihood

$$E(w, \mathbf{h}) = -\log \sigma(\mathbf{u}_{w_o}^\top \mathbf{h}) - \sum_{w_n \in \boldsymbol{N}} \log \sigma(\mathbf{u}_{w_n}^\top \mathbf{h}), \tag{1}$$

where $\boldsymbol{N} = (w_n \sim P(\boldsymbol{W}) | n = 1, ..., K)$ is a set of negative samples (randomly selected words from a noise distribution $P(\boldsymbol{W})$, w_o is the output word, and u_{w_o} is its output vector; $\boldsymbol{h}$ is the output value of the hidden layer: $h = \frac{1}{C}\sum_{C=1..N} \mathbf{v}_{w_c}$ for CBOW and $h = \mathbf{v}_{w_I}$ in the Skip-gram model; $\sigma(x) = 1/(1 + exp(-x))$.

Considering articles a_j, in the ***CBOW*** architecture, the model predicts the current word $w_{j,i}$ from a window of surrounding context words $w_c \in \boldsymbol{C}_{j,i}$. The context is based on bag-of-words hypothesis, so that the order of the words does not influence the prediction. CBOW model optimizes following objective function:

$$\sum_{\boldsymbol{a}_j \in \boldsymbol{D}} \sum_{i=1}^{L_j} E(w_{j,i}, \frac{1}{|\boldsymbol{C}_{j,i}|} \sum_{w_c \in \boldsymbol{C}_{j,i}} \mathbf{v}_{w_c}). \quad (2)$$

According to [21], *CBOW* is faster than *Skip-Gram*, but *Skip-Gram* usually perform better for infrequent words.

4 Wikipedia Category Representation

Wikipedia is a good source of global information. Overall, Wikipedia comprises more than 40 million articles in 301 different languages. Each article references others that describe particular information in more detail. Wikipedia give more information about an article that we might not see at the first moment, such as mentioned links to other articles, or at the end of the article there is a section that describes all categories where current article is belonging. The category system of Wikipedia (see fig. 1) is organized as an overlapping tree [31] of categories[3] with one main category and a lot of subcategories. Every article contains several categories to which it belongs. Categories are intended to group together with pages on similar subjects. Any category may branch into subcategories, and it is possible for a category to be a subcategory of more than one 'parent' category (A is said to be a parent category of B when B is a subcategory of A)[31]. The page editor uses either existing categories, or create one. Generally the user-defined categories are too vague or may not be suitable to use them in our model as a source of global information. Fortunately, Wikipedia provides 25 main topic classification categories for all Wikipedia pages.

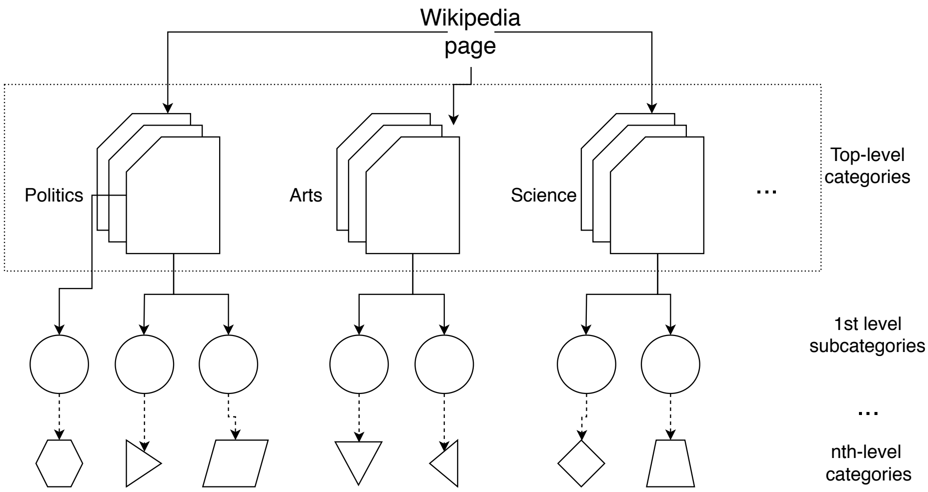

Fig. 1: Wikipedia category system.

[3] https://en.wikipedia.org/wiki/Portal:Contents/categories

For example the article entitled *Czech Republic* has categories *Central Europe*, *Central European countries*, *Eastern European countries*, *Member states of NATO*, *Member states of EU*, *Slavic countries and territories* and others.
Wikipedia categories provide very useful topical information about each article. In our work we use extracted categories to improve the performance of word embeddings. We denote articles as a_j and categories as x_k (see fig. 2).

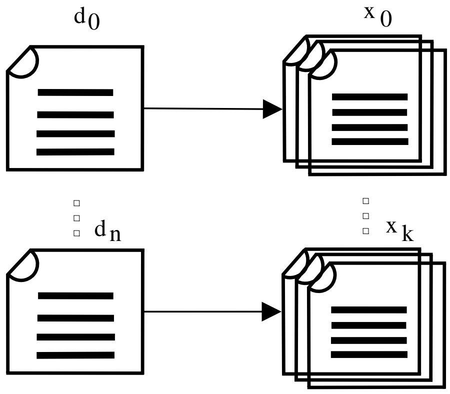

Fig. 2: Document - categories relation.

5 Proposed Model

Some authors tried to extract a more concrete meaning using *Frege's principle of compositionality* [24], which states that the meaning of a sentence is determined as a composition of words. [35] introduced several techniques to combine word vectors into the final vector for a sentence. In [3] experimented with Semantic Textual Similarity, from the tests with words vector composition based on *CBOW* architecture, we can see that this method is powerful to carry the meaning of a sentence.

Our model is shown in Figure 3. We build up the model based on our previous knowledge and believes that Global information might improve the performance of word embeddings and further lead to improvements in many NLP subtasks.

Each article $\boldsymbol{a}_j$ in Wikipedia is associated with the set of categories $\boldsymbol{X}_j$. We represent the category $x \in \boldsymbol{X}_j$ as a real-valued vector $\mathbf{m}_x \in \mathbb{R}^d$.

CBOW model optimize following objective function:

$$\sum_{\boldsymbol{a}_j \in \boldsymbol{D}} \sum_{i=1}^{L_j} E(w_{j,i}, \frac{\sum_{w_c \in \boldsymbol{C}_{j,i}} \mathbf{v}_{w_c} + \sum_{x \in \boldsymbol{X}_j} \mathbf{m}_x}{|\boldsymbol{C}_{j,i}| + |\boldsymbol{X}_j|}) \tag{3}$$

Skip-gram model optimize following objective function

$$\sum_{\boldsymbol{a}_j \in \boldsymbol{D}} \sum_{i=1}^{L_j} \sum_{w_c \in \boldsymbol{C}_{j,i}} E(w_{j,i}, \mathbf{v}_{w_c} + \sum_{x \in \boldsymbol{X}_j} \mathbf{m}_x) \tag{4}$$

Visualization of the CBOW is at following picture:

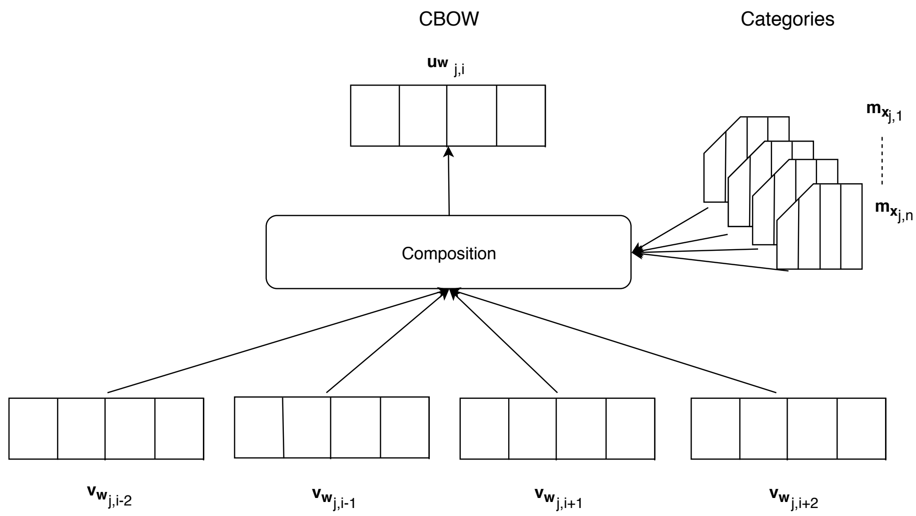

Fig. 3: Architecture of enriched CBOW model with categories.

Visualization of the Skip-gram is at following picture:

We tested with *CBOW* and *Skip-gram* architectures enriched with categories that are shown at pictures 3 and 4. The *CBOW* architecture is generally much faster and easier to train and gives a good performance. The *Skip-gram* architecture is training 10x slower and was unstable during our setup with categories.

5.1 Setup

This article extends our unpublished manuscript that deals with four different types of model architectures and how to incorporate the categories for training the word embeddings, in this work the Czech language has been chosen to test the model with the following setup: Model is initialized with categories that are uniformly distributed. During training the sentence from a training corpora, we add vectors of corresponding categories to actual context window. Motivation of our approach comes from Distributional hypothesis [10] that says: ”words that occur in the same contexts tend to have similar meanings”. If we are training with the categories, we believe they would behave with respect to the Distributional hypothesis.

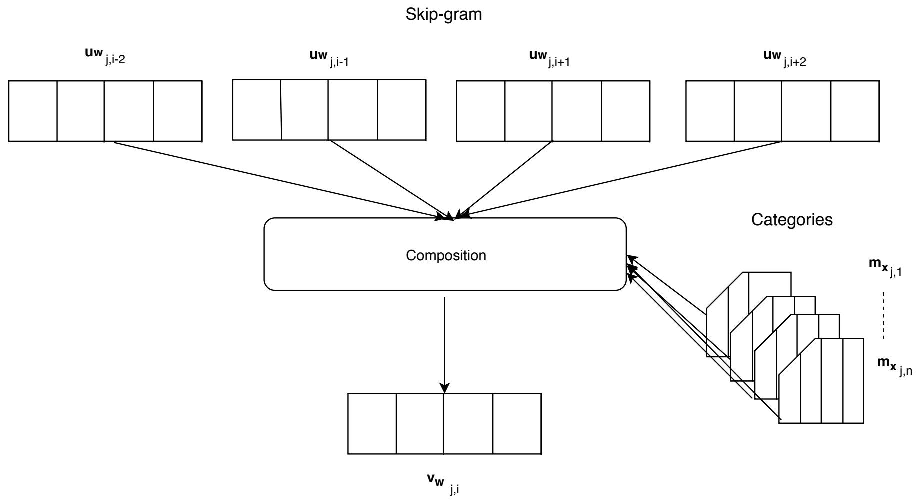

Fig. 4: Architecture of enriched Skip-gram model with categories.

6 Training

	English (dump statistics)
Articles	5,164,793
Words	1,759,101,849
	English (final clean statistics)
Articles	1,554,079
Avg. words per article	437
Avg. number of categories per article	4.69
Category names vocabulary	4,015,918

Table 1: Training corpora statistics. English Wikipedia dump from June 2016.

We previously tested our models on English Wikipedia dump from June 2016[4]. The XML dump consist of 5,164,793 articles and 1,759,101,849 words. We firstly removed XML tags and kept only articles marked with respective id, further we removed articles with less than 100 words or less than 10 sentences. We removed categories that has less than 10 occurrences in between all articles. We have removed the articles without categories. The final corpus used for training consist of 1,554,079 articles. Czech Wikipedia dump comes from March 2017. Detailed statistics on these corpora are shown in Tables 1 and 2. For an evaluation, we experiment with word analogy and a variety of word similarity datasets.

[4] `dumps.wikimedia.org`

- **Word similarity:** These datasets are conducted to measure the semantic similarity between pair of words. For English, these include WordSim-353 [6], RG-65 [28], RW [19], LexSim-999 [12], and MC-28 [23]. For Czech, only two datasets are available and these include RG-65 [15] and WordSim-353 [5]. Both datasets consists of translated word pairs from English and re-annotated with Czech native speakers.
- **Word analogies:** Follow observation that the word representation can capture different aspects of meaning, [21] introduced evaluation scheme based on word analogies. Scheme consists of questions, e.g. which word is related to *man* in the same sense as *queen* is related to *king*? The correct answer should be *woman*. Such a question can be answered with a simple equation: $\text{vec}(\textit{king}) - \text{vec}(\textit{queen}) = \text{vec}(\textit{man}) - \text{vec}(\textit{woman})$. We evaluate on English and Czech word analogy datasets, proposed by [21] and [33], respectively. The word-phrases were excluded from original datasets, resulting in 8869 semantic and 10,675 syntactic questions for English (19,544 in total), and 6018 semantic. 14,820 syntactic questions for Czech (20,838 in total).

	Czech (dump statistics)
Articles	575,262
Words	88,745,854
	Czech (final clean statistics)
Articles	480,006
Avg. words per article	308
Avg. number of categories per article	4.19
Category names vocabulary	261,565

Table 2: Training corpora statistics. Czech Wikipedia dump from June 2016.

6.1 Training Setup

We tokenize the corpus data. We use simple tokeniser based on regular expressions. After model is trained, we keep the most frequent words in a vocabulary ($|\boldsymbol{W}| = 300,000$). Vector dimension for all our models is set to $d = 300$. We always run 10 training iterations. The window size is set 10 to the left and 10 to the right from the center word $w_{j,i}$, i.e. $|\boldsymbol{C}_{j,i}| = 20$. The set of negative samples $\boldsymbol{N}$ is always sampled from unigram word distribution raised to 0.75 and has fixed size $|\boldsymbol{N}| = 10$. We do not use the sub-sampling of frequent words. Process of parameter estimation process is described in detail in [9]. We prefixed categories to be unique in training and not interfering with words during training phase.

fastText is trained on our Wikipedia dumps (see results in Table 3 and 4). *LexVec* is tested only for English, trained on Wikipedia 2015 + NewsCrawl[5], has 7B tokens,

[5] `http://www.statmt.org/wmt14/translation-task.html`

vocabulary of 368,999 words and vectors of 300d. Both (fastText and LexVec) models use character n-grams of length 3-6 as subwords. For a comparison with much larger training data (only available for English), we downloaded *GoogleNews100B*[6] model that is trained using Skipgram architecture on 100 billion words corpus and negative sampling, vocabulary size is 3,000,000 words.

7 Experimental Results and Discussion

We experiment with model defined in Section 5.1 and training Setup from Section 6.1.

		Word similarity				Word analogy		
	Model	**WS-353**	**RG-65**	**MC-28**	**Simlex-999**	**Sem.**	**Syn.**	**Total**
Baselines	fastText - SG 300d wiki	46.12	76.31	73.26	26.78	68.77	67.94	68.27
	fastText - cbow 300d wiki	44.64	73.64	69.67	38.77	69.32	81.42	76.58
	SG GoogleNews 300d 100B	68.49	76.00	80.00	46.54	78.16	76.49	77.08
	CBOW 300d wiki	57.94	68.69	71.70	33.17	73.63	67.55	69.98
	SG 300d wiki	64.73	78.27	82.12	33.68	83.64	66.87	73.57
	LexVec 7B	59.53	74.64	74.08	40.22	80.92	66.11	72.83
	CBOW 300d + Cat	63.20	78.16	78.11	40.32	77.31	68.68	72.13
	SG 300d + Cat	62.55	80.25	86.07	33.54	80.77	71.05	74.93

Table 3: Word similarity and word analogy results on English.

As an evaluation measure for word similarity tasks we use Spearman correlation between system output and human annotations. For word analogy task we evaluate by accuracy of correctly returned answers. Results for English Wikipedia are shown in Table 3 and for Czech in Table 4. These detailed results allow for a precise evaluation and understanding of the behaviour of the method. First, it appears that, as we expected, it is more accurate to predict entities when categories are incorporated.

7.1 Discussion

Distributional word vector models capture some aspect of word co-occurrence statistics of the words in a language [17]. Therefore, these extended models produce semantically coherent representations, if we allow to see shared categories between semantically similar textual data, the improvements presented in Tables 3 and 4 is the evidence of the distributional hypothesis.

Our model on English also outperforms fastText architecture [1] that is a recent improvement of Word2Vec with sub-word information. With our adaptation, the CBOW architecture give similar performance as the Skipgram architecture trained on much

[6] `https://developer.syn.co.in/tutorial/bot/oscova/pretrained-vectors.html`

	Model	Word similarity WS-353	RG-65	MC-28	Word analogy Sem.	Syn.	Total
Baselines	fasttext - SG 300d wiki	67.04	67.07	72.90	49.03	76.95	71.72
	fasttext - CBOW 300d wiki	40.46	58.35	57.17	21.17	85.24	73.23
	CBOW 300d wiki	55.9	41.14	49.73	22.05	52.56	44.33
	SG 300d wiki	65.93	68.09	71.03	48.62	54.92	53.74
	CBOW 300d + Cat	54.31	47.03	49.31	42.00	62.54	58.69
	SG 300d + Cat	62	57.55	64.64	47.03	54.07	52.75

Table 4: Word similarity and word analogy results on Czech.

Table 5: Detailed word analogy results

Table 6: CZ with CBOW and Categories

Type	Baseline	Cat
Antonyms (nouns)	15.72	7.14
Antonyms (adj.)	19.84	46.20
Antonyms (verbs)	6.70	5.00
State-cities	35.80	50.57
Family-relations	31.82	50.64
Nouns-plural	69.44	75.93
Jobs	76.66	95.45
Verb-past	51.06	61.04
Pronouns	11.58	10.42
Antonyms-acjectives	71.43	81.82
Nationalities	20.40	21.31

Table 7: EN with CBOW and Categories

Type	Baseline	Cat.
Capital-common-countries	84.98	88.34
Capital-world	81.78	87.69
Currency	5.56	5.56
City-in-state	62.55	65.22
Family-relations	92.11	90.94
Adjective-to-adverb	25.38	29.38
Opposite	41.67	37.08
Comparative	79.14	78.82
Superlative	59.74	64.50
Present-participle	61.95	65.89
Nationality-adjective	91.39	98.69
Past-tense	63.66	66.59
Plural	74.19	71.67
Plural-verbs	62.33	46.33

larger data. On RG-65 word similarity test and semantic oriented analogy questions in Table 3 it gives better performance. We can see, that our model is powerful in semantics.

There is also significant performance gain on WS-353 similarity dataset and English language. Czech generally perform poorer, because of less amount of data to train and also due to the fact of the language properties. The Czech has free word order and higher morphological complexity that influences the quality of resulting word embeddings, that is also the reason why the sub-word information tends to give much better results. However, our method shows significant improvement in Semantics, where the performance with Czech language has improved twofold (see Table 4).

The individual improvements of word analogy tests with CBOW architecture are available in Table 5. These detailed results allow for a precise evaluation and analyse the behaviour of our model. In Czech language, we see the biggest gain in understanding of category "Jobs". This semantic category is specific to Czech language as it distinguishes between feminine and masculine form of profession. However, we do not see much difference in section ”Nationalities” that also describes countries and masculine versus feminine form of its representatives. We think this might be caused of lack data from Wikipedia. In Czech, we use mostly masculine form in articles when talking about people from different countries. In a section ”Pronouns” that deals with analogy questions such as: *”I,we”* versus *”you, they”*, we clearly cannot benefit from incorporating the categories. The biggest performance gain is as we expected in semantic oriented categories such as: *Antonyms, State-cities and Family-relations*. English gives slightly lower score in *Family-relations* section of analogy corpus. However, as English semantic analogy questions are already hitting correlations above 80% and especially for this section already more than 90%, we believe that we are already hitting the maximal capabilities of machines and humans agreement. This is the reason why we bring up the comparison with highly inflected language. In [33] and [32] has been shown that there is a space for the performance improvement of current state-of-the-art word embedding models on languages from Slavic families. More information about individual section of Czech word analogy corpus is described in [33].

With Czech language, we investigated a drop in performance of the Skip-gram model. This fact might be caused of not enough data for the reverse logic of training the Skip-gram architecture.

8 Conclusion

8.1 Contributions

Our model with global information extracted from Wikipedia significantly outperform the baseline CBOW model. It provide similar performance compared with methods trained on much larger datasets.

We focused on currently widely used CBOW method and Czech language. As a source of global document (respective article) context we used Wikipedia that is available in 301 languages. Therefore,it can be adopted to any other language without necessity of manual data annotation. The model can help to the highly inflected languages such as Czech is, to create word embeddings that perform better with smaller corpora.

8.2 Future Work

We believe that using our method together with sub-word information can have even bigger impact on poorly resourced and highly inflected languages, such as Czech from Slavic family. Therefore, the future community work might lead to integrate our model to the latest architectures such as *fastText* or *LexVec* and improve the performance further from incorporating the sub-word information. Also we suggest to take a look into the other possibilities, how to extract useful information from Wikipedia and how to use it during training - such as references, notes, literature, external links, summary info (usually displayed on the right side of the screen) and others.

We provide the global information data and trained word vectors for research purposes at `https://github.com/Svobikl/global_context/`.

Acknowledgements

This work has been supported by Grant No. SGS-2019-018 Processing of heterogeneous data and its specialized applications. Computational resources were provided by the CESNET LM2015042 and the CERIT Scientific Cloud LM2015085, provided under the programme "Projects of Large Research, Development, and Innovations Infrastructures.

Bibliography

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

[2] Brychcín, T. and Konopík, M. (2015). Latent semantics in language models. *Computer Speech & Language*, 33(1):88–108.

[3] Brychcín, T. and Svoboda, L. (2016). Uwb at semeval-2016 task 1: Semantic textual similarity using lexical, syntactic, and semantic information. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 588–594.

[4] Charles, W. G. (2000). Contextual correlates of meaning. *Applied Psycholinguistics*, 21(4):505–524.

[5] Cinková, S. (2016). Wordsim353 for czech. In *International Conference on Text, Speech, and Dialogue*, pages 190–197. Springer.

[6] Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., and Ruppin, E. (2002). Placing search in context: The concept revisited. *ACM Transactions on information systems*, 20(1):116–131.

[7] Firth, J. R. (1957). A synopsis of linguistic theory, 1930-1955. *Studies in linguistic analysis*.

[8] Gabrilovich, E. and Markovitch, S. (2009). Wikipedia-based semantic interpretation for natural language processing. *Journal of Artificial Intelligence Research*, 34:443–498.

[9] Goldberg, Y. and Levy, O. (2014). word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method. *arXiv preprint arXiv:1402.3722*.

[10] Harris, Z. S. (1954). Distributional structure. *Word*, 10(2-3):146–162.

[11] Hercig, T., Brychcín, T., Svoboda, L., Konkol, M., and Steinberger, J. (2016). Unsupervised methods to improve aspect-based sentiment analysis in czech. *Computación y Sistemas*, 20(3):365–375.

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

[13] Huang, E. H., Socher, R., Manning, C. D., and Ng, A. Y. (2012). Improving word representations via global context and multiple word prototypes. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 873–882. Association for Computational Linguistics.

[14] Konkol, M., Brychcín, T., and Konopík, M. (2015). Latent semantics in named entity recognition. *Expert Systems with Applications*, 42(7):3470–3479.

[15] Krcmár, L., Konopík, M., and Jezek, K. (2011). Exploration of semantic spaces obtained from czech corpora. In *DATESO*, pages 97–107.

[16] Landauer, T. K., Foltz, P. W., and Laham, D. (1998). An introduction to latent semantic analysis. *Discourse processes*, 25(2-3):259–284.

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

[18] Lund, K. and Burgess, C. (1996). Producing high-dimensional semantic spaces from lexical co-occurrence. *Behavior research methods, instruments, & computers*, 28(2):203–208.

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

[20] McNamara, D. S. (2011). Computational methods to extract meaning from text and advance theories of human cognition. *Topics in Cognitive Science*, 3(1):3–17.

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

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

[23] Miller, G. A. and Charles, W. G. (1991). Contextual correlates of semantic similarity. *Language and cognitive processes*, 6(1):1–28.

[24] Pelletier, F. J. (1994). The principle of semantic compositionality. *Topoi*, 13(1):11–24.

[25] Pennington, J., Socher, R., and Manning, C. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

[26] Riordan, B. and Jones, M. N. (2011). Redundancy in perceptual and linguistic experience: Comparing feature-based and distributional models of semantic representation. *Topics in Cognitive Science*, 3(2):303–345.

[27] Rohde, D. L., Gonnerman, L. M., and Plaut, D. C. (2004). An improved method for deriving word meaning from lexical co-occurrence. *Cognitive Psychology*, 7:573–605.

[28] Rubenstein, H. and Goodenough, J. B. (1965). Contextual correlates of synonymy. *Communications of the ACM*, 8(10):627–633.

[29] Salle, A., Idiart, M., and Villavicencio, A. (2016). Matrix factorization using window sampling and negative sampling for improved word representations. *arXiv preprint arXiv:1606.00819*.

[30] Salle, A. and Villavicencio, A. (2018). Incorporating subword information into matrix factorization word embeddings. *arXiv preprint arXiv:1805.03710*.

[31] Shuai, X., Liu, X., Xia, T., Wu, Y., and Guo, C. (2014). Comparing the pulses of categorical hot events in twitter and weibo. In *Proceedings of the 25th ACM conference on Hypertext and social media*, pages 126–135. ACM.

[32] Svoboda, L. and Beliga, S. (2018). Evaluation of croatian word embeddings. In *Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

[33] Svoboda, L. and Brychcin, T. (2016). New word analogy corpus for exploring embeddings of czech words. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 103–114. Springer.

[34] Turney, P. D. and Pantel, P. (2010). From frequency to meaning: Vector space models of semantics. *CoRR*, abs/1003.1141.

[35] Zanzotto, F. M., Korkontzelos, I., Fallucchi, F., and Manandhar, S. (2010). Estimating linear models for compositional distributional semantics. In *Proceedings of the 23rd International Conference on Computational Linguistics*, COLING '10, pages 1263–1271, Stroudsburg, PA, USA. Association for Computational Linguistics.