Improving Word Embeddings for Low Frequency Words by Pseudo Contexts

Fang Li and Xiaojie Wang

School of Computer,
Beijing University of Posts and Telecommunications,
Beijing, China
{golifang,xjwang}@bupt.edu.cn

Abstract. This paper investigates relations between word semantic density and word frequency. A distributed representations based word average similarity is defined as the measure of word semantic density. We find that the average similarities of low frequency words are always bigger than that of high frequency words, when the frequency approaches to 400 around, the average similarity tends to stable. The finding keeps correct with changes of the size of training corpus, dimension of distributed representations and number of negative samples in skip-gram model. It also keeps on 17 different languages. Basing on the finding, we propose a pseudo context skip-gram model, which makes use of context words of semantic nearest neighbors of target words. Experiment results show our model achieves significant performance improvements in both word similarity and analogy tasks.

Keywords: Word Embedding · Low Freuqcy Word

1 Introduction

Representation of word meaning has long been a fundamental task in natural language processing. Traditional methods treat each word a symbol. Distributional representation [20,13,1] represented a word by its context vector, which is high-dimensional and sparse. Distributed representations (i.e. word embeddings) encode words as low-dimensional real-valued vectors. Lots of models, including Collobert and Weston embeddings (C&W) [6], HLBL [17], word2vec [15] and GloVe [18] etc, have been proposed for learning word embeddings. Word embeddings have been widely used in language modeling [2], NER [21], parsing [6] and some other natural language processing tasks.

Meanwhile, there was an extensive work on revealing the properties of distributed representations. [11] demonstrated that skip-gram negative sampling (SGNS) is an implicit weighted matrix factorization of the shifted point mutual information matrix. [12] pointed out that SGNS is an explicit matrix factorization of of the words co-occurrence matrix.

Ideally, the vector space spanned by word embeddings is mainly driven by semantics of words [7]. And the frequency of a word should not be an important

parameter. However, [19] found that word embeddings do contain frequency information, frequency is an important factor on word encoding. [21] evaluated Brown clusters based representations, C&W and HLBL word embeddings on NER task. Experiments showed that most of NER errors are made on words with low frequencies. Brown clusters based representations outperform other distributed representations on low frequency words.

Recently, some models have been proposed to improve embeddings for low frequency words. By exploiting the internal structures of Chinese words, [5] used Chinese characters as features for words with different frequencies. [3] made use of an alphabet based n-gram to improve the embeddings of low frequency words for morphologically rich languages. In generally, these models exploit features that can be shared among different words, thus low frequency word can be enhanced by these features.

However, there are still lots of questions remained for further exploring, such as what is the problem on embeddings of low frequency words? How low frequency hurt embeddings of words? Answers to these questions might provide a principled approach to improve the quality of embeddings for low frequency words.

This paper investigates some of the aforementioned questions. We start the investigation from word semantic density. A distributed representation based word average similarity is firstly defined as a measure for word semantic density. We then find an interesting phenomenon: low-frequency words always have bigger average similarities than those words with high frequency. Further experimental results show that there is a stable relation between average similarities and word frequency. The relation show stability under the different parameters of skip-gram model as well as different languages. Basing on the finding, we propose a pseudo context skip-gram model, which makes used of context words of semantic nearest neighbors of target words. Unlike the feature sharing approch [5,3], this strategy is not language dependent and can be applied in conjunction with other methods simultaneously. Experiment results show our model achieves significant performance improvements in both word similarity and analogy tasks.

2 The Empirical Relation

2.1 Semantic Nearest-neighbors

Let C be a corpus of a language, D is the vocabulary of C, $D = \{w_1, ...w_i, ...w_{|D|}\}$. Let V_{w_i} be the distributed representation of word $w_i, i = 1, ..., |D|$. We denote the similarity between w_i and w_j as

$$sim(w_i, w_j) = cos\_sim(V_{w_i}, V_{w_j}) \tag{1}$$

Where cos_sim denote cosine similarity.

A 154MB English corpus is used to train the skip-gram model[1] by word2vec[2] with its default parameter setting. The similarities between all words in D are

[1] CBOW has similar results. We therefore only give the results of skip-gram.
[2] https://code.google.com/archive/p/word2vec/

then computed by equation (1). Table 1 gives top 10 nearest-neighbors of three words. The three words have different frequencies in the corpus. We can find that the similarities between top 10 nearest-neighbors and word "azeotrope" with frequency=20 are bigger than those of word "invest" with frequency=200, the similarities between top 10 nearest-neighbors and word "invest" are higher than those of word "manual" with frequency=500. i.e., low frequency words are more similar to their nearest-neighbors than that of words with high frequency.

Word	**Frequency**	**Top 10 nearest-neighbors (similarity)**
azeotrope	20	D2O(0.888) eutectic(0.887) A1c(0.887) HDO(0.879) azeotropic(0.875) miscibility(0.873) COF(0.870) hydrophobicity(0.870) Saturation(0.870) SWNT(0.866)
invest	200	recoup(0.783) investing(0.763) repay(0.747) privatize(0.743) invested(0.734) insure(0.720) allocate(0.719) innovate(0.717) exchequer(0.715) approvals(0.715)
manual	500	bookkeeping(0.692) computerized(0.688) pantograph(0.666) Braille(0.664) manuals(0.664) typesetting(0.657) copying(0.643) QWERTY(0.635) automatic(0.627) Procedural(0.624)

Table 1. Three words with their frequencies and top 10 nearest neighbors are shown. Those words are chosen by frequency from low, median and high. The similarity of each neighbor in top 10 nearest neighbors is also given.

Are these some special cases? Or is there a universal law behind? We further inspect it on all words in vocabulary.

2.2 Semantic Density

Let the semantic density of w_i be the average similarity between its word embedding and all other words in D, it is denoted by $avg\_sim(w_i)$ and calculated by equation (2).

$$avg\_sim(w_i) = \frac{1}{|D|} \sum_{w_j \in D} sim(w_i, w_j) \tag{2}$$

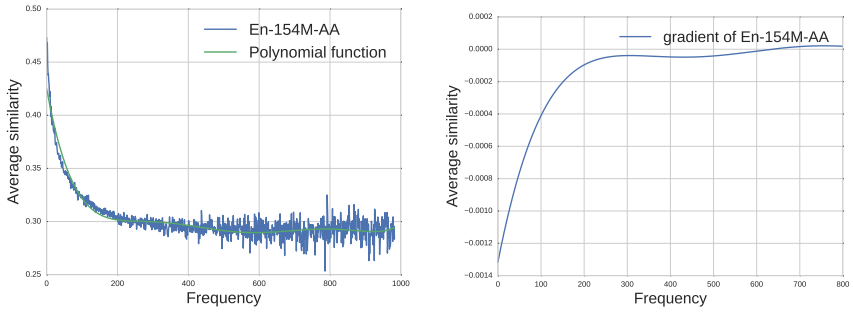
Let f_{w_i} be the frequency of word w_i, we then define the semantic density of the words with frequency=K. Given a frequency K, M words are uniformly sampled from the set of all words with frequency=K (For simplification, M words instead of all words are sampled. We find $M = 50$ is enough in experiments). Let S_K denotes the set of these M words, $AvgS_K$ denotes average similarity of words with frequency=K, is then calculated by (3)

$$AvgS_K = \frac{1}{M} \sum_{w_i \in S_K} avg\_sim(w_i) \tag{3}$$

$AvgS_K$ is computed for K range from 5 to 1000 in a 154MB English corpus. Figure 1(a) is the curve of $AvgS_K$ about K. As depicted in the figure, when

K increases, $AvgS_K$ declines, i.e., low frequency words have larger $AvgS_K$ than those with high frequency. low frequency words are closer to other words in average, they have bigger semantic density. More frequent words have lower $AvgS_K$ and lower semantic density. But when frequency reaches at 400 around, the curve tends to stable. i.e., words with big enough frequencies will have stable semantic density.

In order to inspect the change rate of the average similarity , we fit the K-$AvgS_K$ by a polynomial function, we find that a 5th order polynomial function $y = [-2.99, 8.63, -9.38, 4.75, -1.14, 3.88]^T[(10^{-3}x)^5, (10^{-3}x)^4, (10^{-3}x)^3, (10^{-3}x)^2, 10^{-3}x, 1(10^{-3}x)^5]$ fits the curve well, the polynomial function is also illustrated in Figure 1(a). We then compute the gradient of the polynomial function. The gradient curve is presented in Figure 1(b). These two figures demonstrate that as K increases, $AvgS_K$ decreases, but the rate of change continues to decline, when frequency is near about 400, the similarity reaches a stable value.



(a) (b)

Fig. 1. The average similarity curve and its gradient curve are shown. Left: The average similarity curve on the 154MB english corpus (En-154M-AA) and its polynomial fitting curve are shown. Right: The gradient of the polynomial fitting curve are given.

3 Invariance of the Relation

This section investigates the invariance of our proposed relation. We figure out if this relation holds for various settings for training word embeddings, including several important hyper-parameters in word embeddings learning model (skip-gram is considered in this paper) and languages. Details are described as follows.

- Dose this relation hold when trained on different but sufficiently large corpus size?
- Dose this relation hold with different dimensions?
- Dose this relation hold with different languages?
- Dose this relation hold with other hyper-parameters?

3.1 Corpus Size

Three Chinese corpora with the size of 300MB, 5.6GB and 8.4GB are used for training word embeddings respectively. The K-$AvgS_K$ curves for different corpora are shown in Figure 2(a). They have similar shapes but with different average similarity. Word embeddings trained on a large corpus has a lower average similarity than that on a small corpus, suggest that the word embeddings trained on a big corpus are more distinguishable than that on a small corpus. A word will become more distinguishable when it occurs more frequently. However, according to Zipf's law [24], even with a large corpus the low frequency words still exists. So do large semantic density of words. And the gradients of all curves tend to be zeroes when frequency nearly arrives at 400.

3.2 Dimension of Word Embeddings

Word embeddings with different dimensions from 100 to 1,000 (by step size of 100) are obtained. K-$AvgS_K$ curves for different dimensions are illustrated in Figure 2(b). The legend "zh_100" means that the language is Chinese and the dimension is 100.

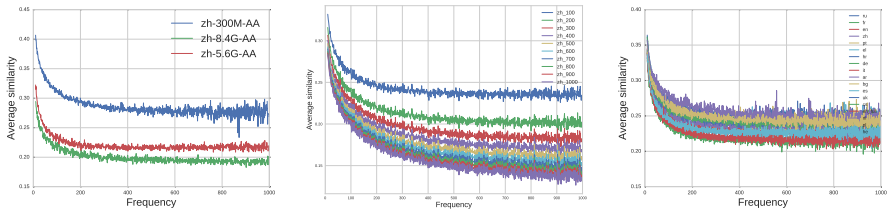
The figure shows that the shape of curves does not significantly change with the dimension. But embeddings with a larger dimension has lower average similarity and semantic density. That comply with the intuition that as the dimension grows, word embeddings become more sparse. As with other situations, the gradients of different curves also tends to zeroes when the frequency nearly approaches to 400.

3.3 Different Languages

So far, we have investigated the hypothesis on English and Chinese. How about other languages? We train word embeddings on seventeen languages. All corpora for different languages are available in wikipedia[3]. Two different English corpora (En and En_full) are used in this experiment.

The K-$AvgS_K$ curves for seventeen languages are presented in Figure 2(c). The curves for all languages are similar. Sepcifically, they go down with the increasing of the frequency. And approach to stable values when the frequency equals 400 around. Different languages have different stable values. Among all those languages, Dutch has the biggest stable value, while French has the smallest one. The gradient of all K-$AvgS_K$ curves also tend to 0 when frequency is near 400. From this figure, we draw the same conclusion as above that gradients of all K-$AvgS_K$ curves tend to be zeroes when the frequency is near 400. This implies that the relations between the frequency K and $AvgS_K$ hold. And 400 is the boundary of low frequency words and the other words for all seventeen languages.

[3] https://dumps.wikimedia.org/backup-index.html



(a) (b) (c)

Fig. 2. Average similarities impacted by three factors, corpus size, embeddings dimension and languages are shown, respectively. (a) Average similarities with frequencies in three different sizes of corpora. (b) Average similarities with frequencies in different embeddings dimensions from 100 to 1,000 with the step size of 100. (c) Average similarities for seventeen languages are reported. and each language is represented by its ISO code.

3.4 General Discussion

Except for three parameters above, we have verified that the hypothesis is also invariant for other parameters, such as number of negative samples, rejection threshold of models. Due to the space limit we do not present here.

Our hypothesis, gives hints on how to improving word embeddings, especially for low frequency words. We will propose an efficient way in next section.

On the other hand, to explore reasons behind the linguistic phenomenon is also important. Polysemy might be a part explanation for the phenomenon. Since frequent words normally be more polysemous, therefore might have lower average similaries than those of low frequency words. Nevertheless, the phenomenon gives us more information. The invariance on different model parameters and different languages, decrease of $K\text{-}AvgSk$ curves stops when frequency beyond 400, all these cannot be simply explained by polysemy.

4 Pseudo-Context Word Embedding

To improving word embeddings, we propose a strategy called "pseudo context" to get much more training data for low frequency words by making use of semantic nearest-neighbors of them. The strategy can be easily incorporated into various existing word embedding models. In this paper, we take skip-gram as an example to introduce the pseudo context based skip-gram (PCSG).

Let corpus $C = w_1, w_2, ..., w_N$ be a sequence of words. V is the vocabulary of all words in C. The objective function of skip-gram model is to maximize the log-likelihood of a center word w_n predicting its context word. The equation is shown in 4).

$$\mathcal{L} = \sum_{n=0}^{N} \sum_{c \in C_n} \log P(w_c | w_n) \tag{4}$$

where the context $C_n = n-L, ..., n-1, ..., n+1, ...n+L$ is the set of index of words within the sized L window of target word w_n. Evaluating the conditional probability $P(w_c|w_n)$ is computationally expensive, which involves the normalized probability of w_n predicting w_c over all other words in the vocabulary. Thus, skip-gram model employs negative sampling to approximate this probability. Its objective function is as follows,

$$\mathcal{L} = \sum_{n=0}^{N} \sum_{c \in C_n} l(w, c) \tag{5}$$

$$l(w, c) = \log \frac{1}{1 + e^{-V_c \cdot V_w}} + k \sum_{c' \sim P_D} \frac{1}{1 + e^{V_{c'} \cdot V_w}} \tag{6}$$

where V_w is the word vector of word w, and c is the word in the context of w. c' is a sample drawn form the distribution of negative words P_D. k is the number of negative samples, which is a trade-of between approximating accuracy and computational complexity.

In PCSG, different objective functions are used for high frequency words and low frequency words. A word is took as low frequency when its frequency is lower than a given threshold T. Objective function for high frequency words remain as in (5). For low frequency word w_m, a different object function is defined. We first construct a similar words set S_m for w_m. The set S_m consists of top N semantic nearest neighbor words of w_m. For any word w_s in this set, we take context words of it as context words of w_m as well. For a context word w_c of w_s, it may not be a true context word of w_m. However, since the two words w_s and w_m are similar, they tend to have similar context according to distributional hypothesis [10]. We call w_c as a **pseudo context** word of w_m. During the training, when w_m is updated, a word w_s from S_m is uniformly sampled, the objective is to maximize the probability of w_s predicting w_c. Equation (4) is therefore replaced by equation (7).

$$\mathcal{L} = \sum_{n=0}^{N} \sum_{c \in C_n, s \in S_n} \log P(w_c|w_n) + \log P(w_c|w_s) \tag{7}$$

Negative sampling method can also be applied. The corresponding objective function for low frequency word is (8).

$$\mathcal{L} = \sum_{n=0}^{N} \sum_{c \in C_n, s \in S_n} l(w, c) + l(s, c) \tag{8}$$

In which the two terms $l(w, c)$ and $l(s, c)$ are defined in equation (6).

5 Experiments

Implementation details. Our baselines are skip-gram (SG) model from word2vec program[4] and CWE+P model from CWE program[5]. Wikipedia corpus is used to train word vector for different languages. All the models are trained with 5 negative samples with rejection threshold 10^{-3} and keep words appearing at least 5 times. The dimension of word vectors is set to 100. By introducing pseudo context for low frequency words, our method increases the computational complexity by approximately 30% in English corpus. However, our implementation is well optimized, about 1.7 times faster than the word2vec implementation of skip-gram. Our code will be available online[6].

We compare the performance of different models on two word based tasks, word relatedness and analogy reasoning.

	English							Chinese			
	Semantic						Syntactic	Semantic			
	total	capital-common-countries	capital-world	currency	city-in-state	family	total	total	capital-common-countries	city-in-state	family
SG	38.29	64.29	41.75	2.94	15.38	72.22	54.25	67.98	70.33	76.57	48.48
PCSG	45.59	73.81	53.33	2.94	20.84	63.40	54.54	69.95	70.99	80.57	52.27
CWE	/	/	/	/	/	/	/	66.01	67.03	78.29	46.21
Δ	7.30	9.52	11.58	0	5.46	-8.82	0.29	1.96	0.66	2.28	3.79

Table 2. Evaluation accuracies($\times$100) on analogy task. For semantic questions, we report the results on different sections and the total dataset.

Word analogy task. An analogy question is like "France is to Paris as Italy to X". In this example, the word X is predicted by finding a word whose vector has the highest cosine similarity with vector $V(France) - V(Italy) + V(Paris)$. Here "Rome" is the correct answer.

Two datasets, google analogy dataset [16] on English and the one from [5] on Chinese are used in our exmperiments. Analogy questions in English dataset are divided into semantic and syntactic questions. Semantic questions contain five sections. The example given above is from the "capital-common-countries" section of semantic question. An example of syntactic question is "free is to freely as usual is to X", where the answer is "usually". Chinese does not contain the same morphological information, so only semantic question is provided, which contains three sections.

Accuracies for sections and for the total dataset are reported in Table 2. In Chinese, besides SG model, CWE model [5] is also used for comparison. By

[4] https://code.google.com/archive/p/word2vec/
[5] https://github.com/Leonard-Xu/CWE
[6] https://github.com/mklf/PCWE

training word and character embeddings together, CWE model also uses the information of Chinese characters. The results are reported in Table 2. We see that (1) PCSG substantially outperforms the other models on semantic questions in both English and Chinese datasets. (2) There is minor change on performances of syntactic questions. We infer the reason is that nearest neighbor words in S_n are semantically similar words of w_n, which have nearly no syntactic information. If syntactic information can be incorporated in the nearest neighbor word selection phase, for example, filtering out the subset of words with same prefix or suffix in S_n in morphologically rich languages, syntactic performance may also be improved. The detailed implementation is left for future work.

	English					Chinese				German	
	WS353	RW		MEN		PKU500		C240	C297	ZG222	Gur350
		all	< 400	all	< 400	all	< 400				
SG	67.67	39.19	31.91	62.50	53.91	35.50	43.67	54.65	54.74	39.48	61.39
PCSG	69.36	42.59	36.40	65.22	57.33	36.86	48.31	56.85	57.03	44.80	62.71

Table 3. Evaluation results on various datasets ($\rho \times 100$). For datasets RW, MEN and PKU500, the correlation coefficient only on low frequency (< 400) words are also measured.

Word relatedness task. This task contains a set of word pairs. The cosine distance of word vectors is computed to score the similarity between a pair of word. Then the spearman correlation coefficient ρ between scores by vector of words and human judgments are then obtained. A higher coefficient for word vectors means a better performance.

Several publicly available word similarity datasets in three languages are used. They consist of three English datasets, WordSim353 (WS353) [8], RareWords (RW) [14], MEN [4], three Chinese datasets PKU500 [22], CWE240(C240), CWE297(C297) [5] and two German datasets ZG222 [9], Gur350 [23]. Among those datasets, RW, MEN and PKU500 have 183, 732, 47 low frequency (frequency < 400) word pairs respectively. Whereas the other datasets contain less than 20 low frequency word pairs.

The spearman correlation coefficient ρ for different models and different datasets are shown in Table 3. For datasets RW, MEN and PKU500, the correlation coefficient only on low frequency words are also measured. We can find that (1) PCSG outperforms SG on all languages and datasets by a margin of 2% 5%. (2) More improvements are achieved for low frequency words on RW, MEN and PKU500. (3) The pseudo context strategy can be applied to di erent languages. Evidently, introducing pseudo context helps to build better word vectors, especially for low frequency words. The results show word vectors trained by PCSG actually include more semantic information by making use of pseudo context.

6 Conclusion and Future Works

One of the goals of computational linguistics is to find interesting linguistic phenomena and reveal their natures in a computational way.

This paper finds some interesting linguistic phenomena based on distributed representations of words. A hypothesis on the relation between distributed representation based average similarities and the frequency of words is proposed. That is low frequency words have larger average similarities. As the frequency increases, the average similarity decreases. When the frequency reaches to 400 around, the average similarity becomes stable. Experimental results show that the relation holds on word embeddings trained by different sizes of corpora and parameter settings. Also, it holds on different languages as well.

Basing on those findings, we propose a pseudo context strategy for low-frequent words. By applying this strategy to skip-gram model, we achieve significant improvement on both word relateness and analogy tasks, especially on low-frequent words.

Acknowledgments. This paper is supported by 111 Project (No. B08004)NSFC (No.61273365) , Beijing Advanced Innovation Center for Imaging Technology, Engineering Research Center of Information Networks of MOE, and ZTE.

References

1. Baroni, M., Lenci, A.: Distributional memory: A general framework for corpus-based semantics. Computational Linguistics 36(4), 673–721 (2010)
2. Bengio, Y., Ducharme, R., Vincent, P., Jauvin, C.: A neural probabilistic language model. Journal of machine learning research 3(Feb), 1137–1155 (2003)
3. Bojanowski, P., Grave, E., Joulin, A., Mikolov, T.: Enriching word vectors with subword information. arXiv preprint arXiv:1607.04606 (2016)
4. Bruni, E., Boleda, G., Baroni, M., Tran, N.K.: Distributional semantics in technicolor. In: Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1. pp. 136–145. Association for Computational Linguistics (2012)
5. Chen, X., Xu, L., Liu, Z., Sun, M., Luan, H.: Joint learning of character and word embeddings. In: Proceedings of IJCAI. pp. 1236–1242 (2015)
6. Collobert, R., Weston, J.: A unified architecture for natural language processing: Deep neural networks with multitask learning. In: Proceedings of the 25th international conference on Machine learning. pp. 160–167. ACM (2008)
7. Faruqui, M., Tsvetkov, Y., Rastogi, P., Dyer, C.: Problems with evaluation of word embeddings using word similarity tasks. arXiv preprint arXiv:1605.02276 (2016)
8. Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., Ruppin, E.: Placing search in context: The concept revisited. In: Proceedings of the 10th international conference on World Wide Web. pp. 406–414. ACM (2001)
9. Gurevych, I.: Using the structure of a conceptual network in computing semantic relatedness. In: International Conference on Natural Language Processing. pp. 767–778. Springer (2005)
10. Harris, Z.S.: Distributional structure. Word 10(2-3), 146–162 (1954)

11. Levy, O., Goldberg, Y.: Neural word embedding as implicit matrix factorization. In: Advances in neural information processing systems. pp. 2177–2185 (2014)
12. Li, Y., Xu, L., Tian, F., Jiang, L., Zhong, X., Chen, E.: Word embedding revisited: A new representation learning and explicit matrix factorization perspective. In: Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI. pp. 25–31 (2015)
13. Lund, K., Burgess, C.: Producing high-dimensional semantic spaces from lexical co-occurrence. Behavior Research Methods, Instruments, & Computers 28(2), 203–208 (1996)
14. Luong, T., Socher, R., Manning, C.D.: Better word representations with recursive neural networks for morphology. In: CoNLL. pp. 104–113 (2013)
15. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781 (2013)
16. Mikolov, T., Yih, W.t., Zweig, G.: Linguistic regularities in continuous space word representations. In: Hlt-naacl. vol. 13, pp. 746–751 (2013)
17. Mnih, A., Hinton, G.E.: A scalable hierarchical distributed language model. In: Advances in neural information processing systems. pp. 1081–1088 (2009)
18. Pennington, J., Socher, R., Manning, C.D.: Glove: Global vectors for word representation. In: EMNLP. vol. 14, pp. 1532–1543 (2014)
19. Schnabel, T., Labutov, I., Mimno, D., Joachims, T.: Evaluation methods for unsupervised word embeddings. In: Proc. of EMNLP (2015)
20. Schutze, H.: Dimensions of meaning. In: Supercomputing'92., Proceedings. pp. 787–796. IEEE (1992)
21. Turian, J., Ratinov, L., Bengio, Y.: Word representations: a simple and general method for semi-supervised learning. In: Proceedings of the 48th annual meeting of the association for computational linguistics. pp. 384–394. Association for Computational Linguistics (2010)
22. Wu, Y., Li, W.: Overview of the nlpcc-iccpol 2016 shared task: Chinese word similarity measurement. In: International Conference on Computer Processing of Oriental Languages. pp. 828–839. Springer (2016)
23. Zesch, T., Gurevych, I.: Automatically creating datasets for measures of semantic relatedness. In: Proceedings of the Workshop on Linguistic Distances. pp. 16–24. Association for Computational Linguistics (2006)
24. Zipf, G.K.: Human behavior and the principle of least effort (1950)