Approximating Word Ranking and Negative Sampling for Word Embedding

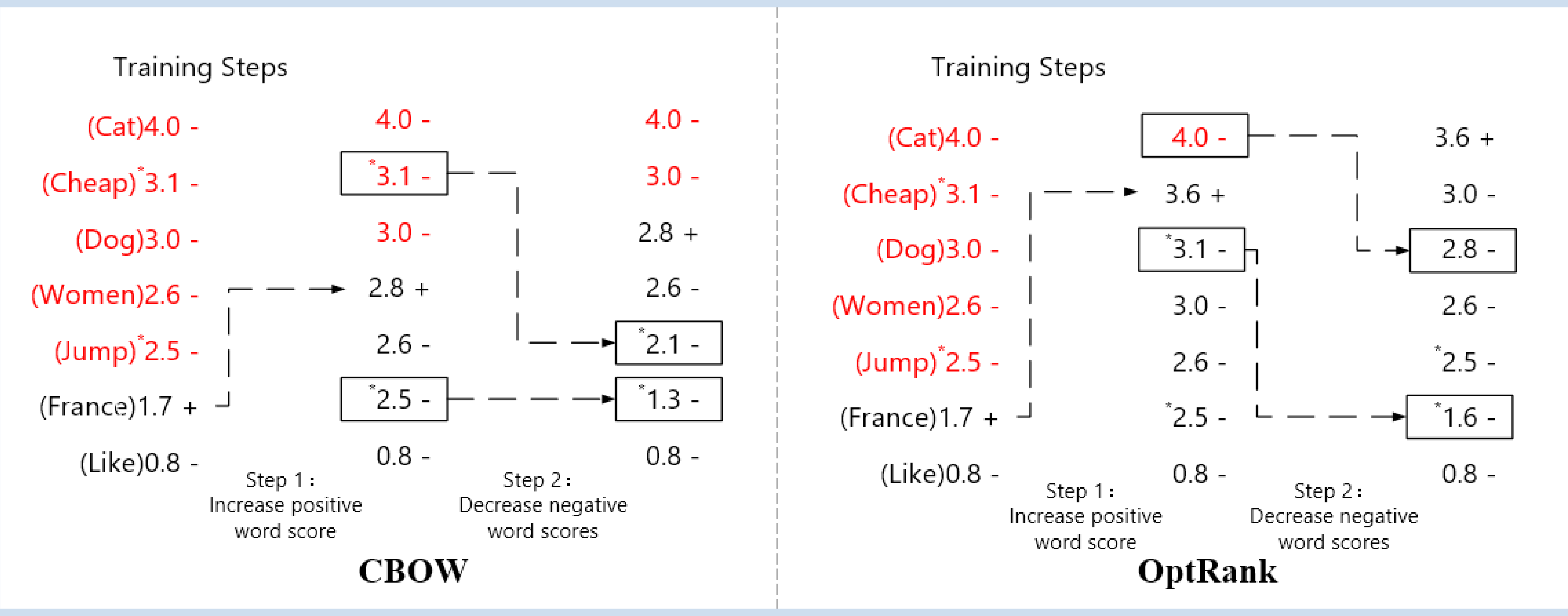
Guibing Guo, Shichang Ouyang, Fajie Yuan, Xingwei Wang
{guogb,wangxw}@swc.neu.edu.cn, 1701282@stu.neu.edu.cn, f.yuan.1@research.gla.ac.uk

Introduction

- CBOW (Continuous Bag-Of-Words) is one of the most commonly used techniques to generate word embeddings in various NLP tasks. However, it fails to reach the optimal performance due to uniform involvements of positive words and a simple sampling distribution of negative words.
- We formalize word embedding as a ranking problem and propose a new model called OptRank which weighs the positive words by their ranks such that highly ranked words have more importance, and adopts a dynamic sampling strategy to select informative negative words.
- In addition, an approximation method is designed to efficiently compute word ranks. Empirical experiments show that OptRank consistently outperforms its counterparts on a benchmark dataset with different sampling scales, especially when the sampled subset is small.

OptRank Model

Using a figure to illustrate the process procedure (steps) of the CBOW and our OptRank models.



OptRank vs CBOW In order to distinguish the positive word 'France'(+) from other negative words(−). Both models will train the word vector in two steps. The word denoted with symbol '*' means that it is a popular word in the corpus. And the red words are the negative words which have higher relevance scores than the positive word.

Step1 : CBOW will increase the relevance score of the positive word by the gradient values from 1.7 to 2.8. The score is still smaller than some other negative words, but the OptRank model can increase the ranking score of the positive word to a larger extent with the help of item ranks.

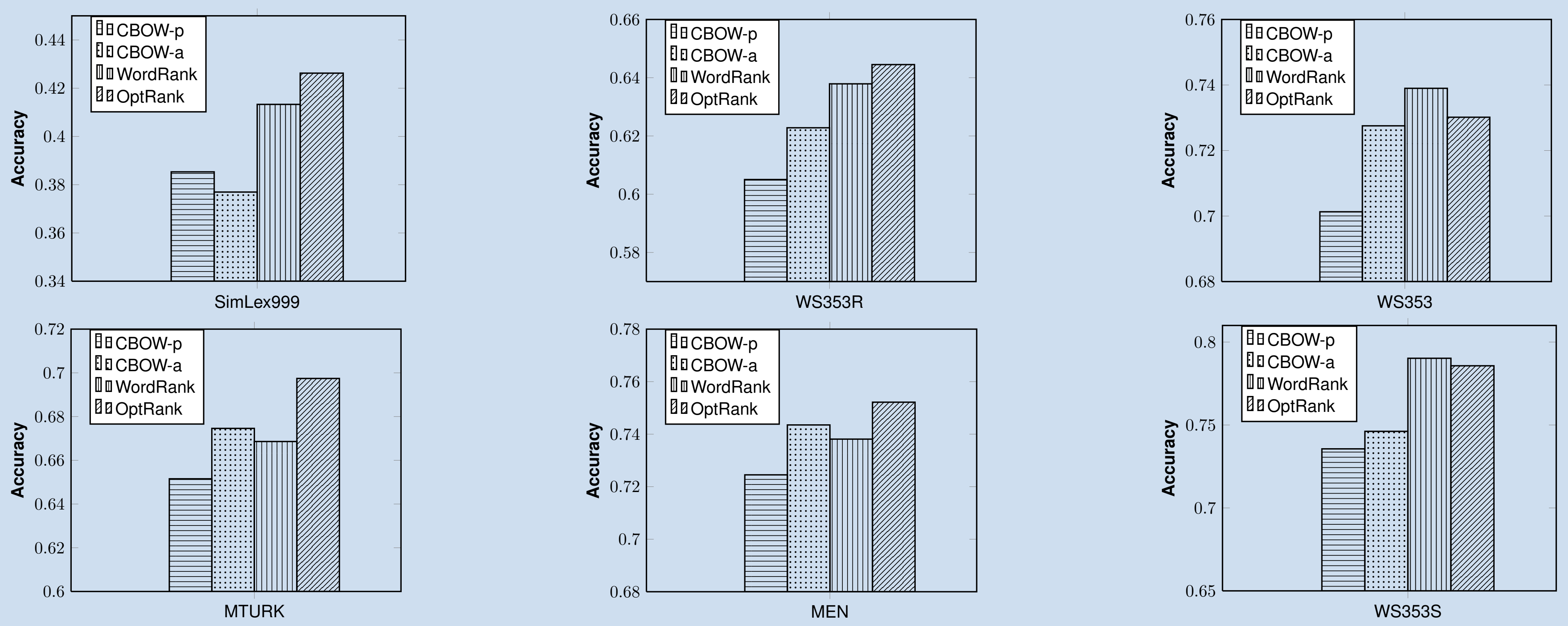
Step2 : CBOW will sample some popular words (e.g., Cheap) denoted by '*' as the negative words, leading to a better yet suboptimal ranking list after step 2. However, OptRank adopts dynamic sampling to find an informative negative example (i.e., word cat) and decrease its ranking score. After that, the positive word will be ranked highest in this intuitive example.

Experimental Results

The word analogy task is to answer the questions in the form of "a is to b as c is to ?" and the word similarity task is to calculate the consine similarity between two relevant words. The best performance of each word embedding model in two testing tasks when the training datasets are relatively small:

Corpus	Word Analogy			Word Similarity		
	CBOW	WordRank	OptRank	CBOW	WordRank	OptRank
128M	0.364	0.415	0.437	0.622	0.633	0.637
256M	0.438	0.518	0.542	0.634	0.651	0.654
512M	0.543	0.642	0.658	0.643	0.657	0.675
1G	0.660	0.647	0.675	0.641	0.670	0.661
2G	0.691	0.685	0.718	0.647	0.665	0.672

The best performance of each word embedding model (trained on 14G Wiki2017) for the task of word similarity:



Effective Learning Scheme

Next we present a learning scheme to effectively train our proposed model. Specifically, for each given training word-context example (w_p, c), we need to compute the ranking value of $rank(w_p, c)$, the exact value of which requires an exhaustive search in the whole word space. It is thus a very time-consuming step, and will become prohibitively expensive when being applied in a large-scale dataset.

To reduce the computational cost, we devise an approach to approximate the rank value by repeated sampling. Specifically, given a training example (w_p, c), we repeatedly sample a negative word from the corpus W until we obtain an expected word w_n that satisfy the requirement given by $v_c^\top v_{w_n} + \varepsilon > v_c^\top v_{w_p}$. That is, the ranking score of the negative word is greater than that of a positive word with tolerance value ε. Let k denote the number of sampling trials to retrieve a proper negative word. This number k follows a geometric distribution with parameter $p = \frac{rank(w_p,c)}{|corpus|} = \frac{rank(w_p,c)}{|W|}$. Then, the expectation of a geometrical distribution with parameter p is $\frac{1}{p}$, i.e., $k \approx \frac{1}{p} = \frac{|W|}{rank(w_p,c)}$. Thus, we can estimate the rank value by $rank(w_p, c) \approx \frac{|W|}{k}$.

Conclusion

In this paper, we view word embedding as a ranking problem and then analyze the main disadvantage of CBOW model that it does not consider the relation between positive and negative words. This easily results in incorrect ranks of words, and produces suboptimal embeddings during training. Thus, we proposed a novel rank model which learns word representations not only by weighting positive words, but also by oversampling informative negative words. Other models typically only pay attention to one of them. The experimental results show that the OptRank model consistently yields a much better performance than the CBOW-p and CBOW-a model, and beats WordRank on many datasets. And OptRank is dominant on the word analogy task for all cases. Moreover, by using an effectively learning scheme, we reduce the computational cost of the OptRank, which makes it become a more practising model. These attributes significantly enable OptRank to achieve good performance even if the training datasets are limited.

Contact