

Zhong-lin YE, Hai-xing ZHAO, 2018. Syntactic word embedding based on dependency syntax and polysemous analysis. *Frontiers of Information Technology & Electronic Engineering*, 19(4):524-535.

<https://doi.org/10.1631/FITEE.1601846>

Syntactic word embedding based on dependency syntax and polysemous analysis

Key words: Dependency-based context; Polysemous word representation; Representation learning; Syntactic word embedding

Corresponding author: Hai-xing ZHAO

E-mail: h.x.zhao@163.com

 ORCID: <http://orcid.org/0000-0002-2429-3325>

Motivation

Most existing word embedding models have the following problems:

1. For those models based on bag-of-words contexts, the structural relations of sentences are completely neglected.
2. Each word uses a single embedding, which makes the model indiscriminative for polysemous words.
3. Word embeddings easily tend to contextual structure similarity of sentences.

Main idea

1. A polysemous tagging algorithm is used for polysemous representation by the Latent Dirichlet Allocation (LDA) algorithm.
2. Symbols '+' and '-' are adopted to indicate the directions of the dependency syntax.
3. Stopwords and its dependencies are deleted.
4. Dependency 'skip' is applied to connect indirect dependencies.
5. Dependency-based contexts are inputted to a word2vec model.

Method

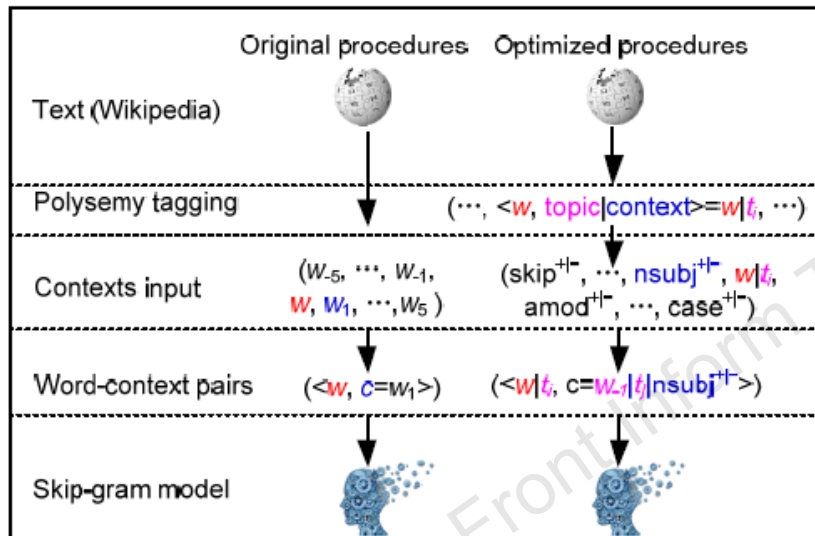


Fig. 1 Comparison of the original procedures and optimized procedures while training the syntactic word embedding

Algorithm 1 Syntactic word embedding generation

Input: entire corpus

Output: polysemous language model

```

1: corpus ← Wikipedia text
2: for  $s \leftarrow$  corpus do
3:    $s \leftarrow$  denoising( $s$ )
4:   list.add( $s$ )
5: end for
6: corpus ← polysemytagging(corpus)
7: for sentence $i$  ← corpus do
8:   for word $j$  ← sentence $i$  do
9:     cxt ← contextcapturing(word $j$ , sentence $i$ )
10:    map.put(word $j,i$ , cxt)
11:   end for
12: end for
13: polysemymodel ← skip-gram(map)
  
```

Major results

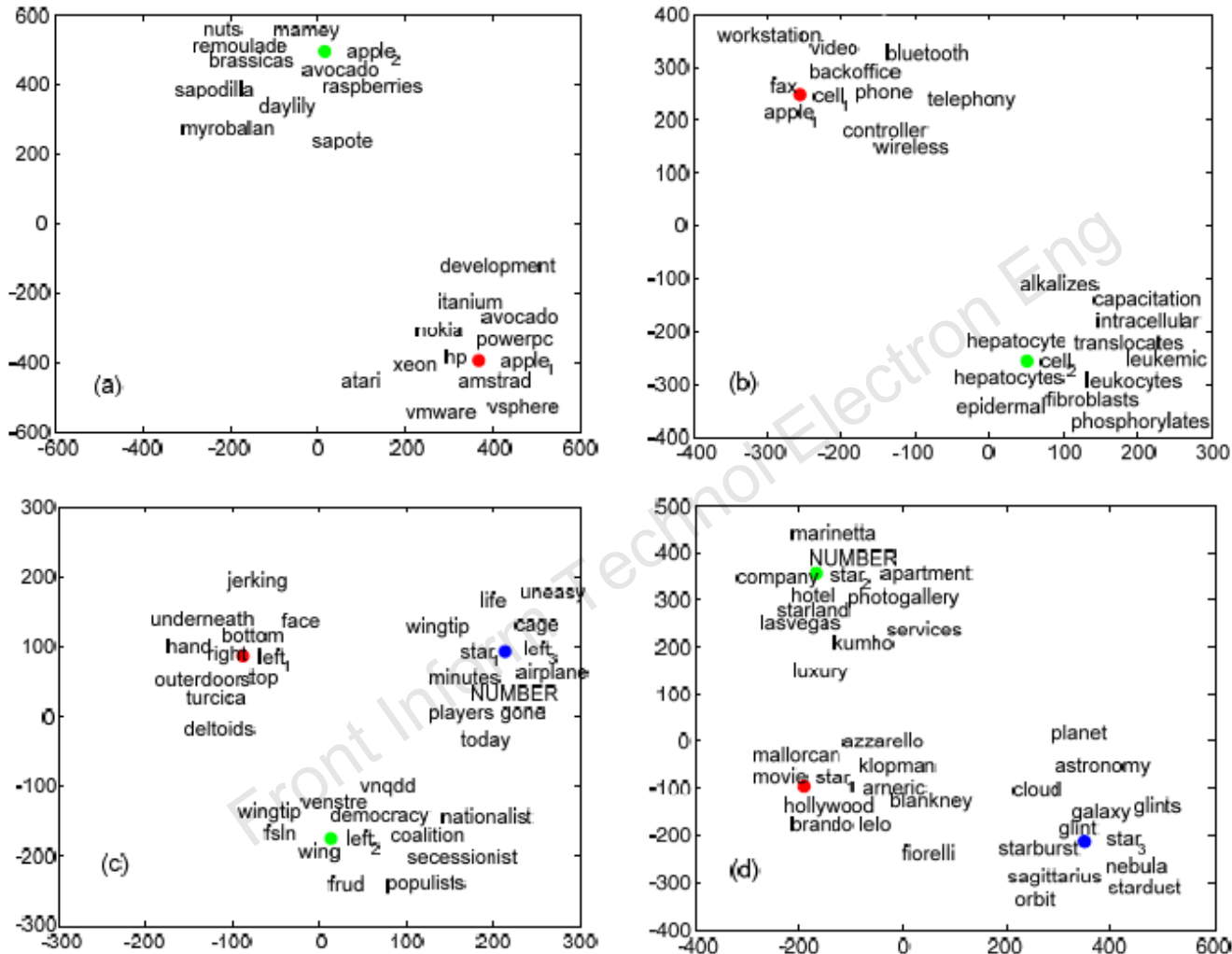


Fig. 3 The neighboring words and 2D visualization: (a) results of polysemy tagging and 2D visualization for apple; (b) results of polysemy tagging and 2D visualization for cell; (c) results of polysemy tagging and 2D visualization for left; (d) results of polysemy tagging and 2D visualization for star

Polysemy has multiple senses; therefore, it exists in multiple communities. Different communities show a same clustering phenomenon. The x -axis and y -axis represent the length of the embedding in the 2D space. References to color refer to the online version of this figure

Major results

Table 6 The performance of different algorithms on five datasets

Algorithm	Absolute value of the Hellinger distance				
	RG	RW	SCWS	SimLex	WS
Skip-gram	3.996	32.986	24.453	16.677	10.692
CBOW	4.092	32.831	25.000	16.601	11.069
Huang et al. (2012)	4.161	21.487	18.710	21.290	11.394
GloVe	5.087	34.735	21.654	14.665	10.249
SWE	3.769	19.993	17.435	14.417	7.687
SWE+TF-IDF	3.919	24.930	21.806	17.641	9.901

RG: Rubenstein and Goodenough; RW: rare word; SCWS: sentential context word similarity; SimLex: SimLex-999; WS: word similarity-353

Conclusions

1. We propose a polysemous language model to generate multiple word embedding for polysemous words.
2. We generalize the skip-gram model with optimized dependency-based contexts, and our model produces markedly different embedding.
3. Our model can generate state-of-the-art word embedding on word similarity tasks. The proposed SWE model is less topical and exhibits more functional similarity compared with other embedding models.