何佳芳
106065502
17 June 2018

Synset expansion

Abstract

The propose is to find out synonym of one giving word according to its sense by Google News pre-trained word2vec model.

We can get synonym of a giving word by Google News pre-trained word2vec model. For example, we input word ‘plant’, and the synonym after calculating by word2vec model is the following.

However, there will be some problem:

1. The most similar synonym usually has a high frequency, then we can get other synonym has lower frequency.
2. The synonym we get is display without considering its part of speech and sense.

```
>>> word2vectors.most_similar('plant', tops = 10)
```

```
('plants', 0.8109676837921143),
('Plant', 0.7019316554069519),
('factory', 0.670879483222961 4),
('paperboard_mill', 0.5969303250312805),
('containerboard_mill', 0.58636909723281 86),
('factories', 0.57695072889328),
```

It is much appropriate if showing the synonym one by one depends on each sense.

Method

1. Input a word and get all information of each senses from WordNet.

Hypernyms	Lemmas	Definition
building_complex (建築物)	**industrial_plant plant works**	buildings for carrying on industrial labor
organism (植物)	**plant_life flora plant**	(botany) a living organism lacking the power of locomotion
actor (演員)	**actor plant**	an actor situated in the audience whose acting is rehearsed but seems spontaneous to the audience
contrivance (詭計)	**plant contrivance**	something planted secretly for discovery by another

2. Extract hypernym and lemmas as its **positive_synset** of each sense.
3. Take hypernym from other senses as **negative_synset**.

Sense	Positive synset	Negative synset
organism (植物) (botany) a living organism lacking the power of locomotion	**plant_life flora plant organism**	**plant actor contrivance**

4. Transform every word in **positive_synset** and **negative_synset** into a 300-dimension vector.

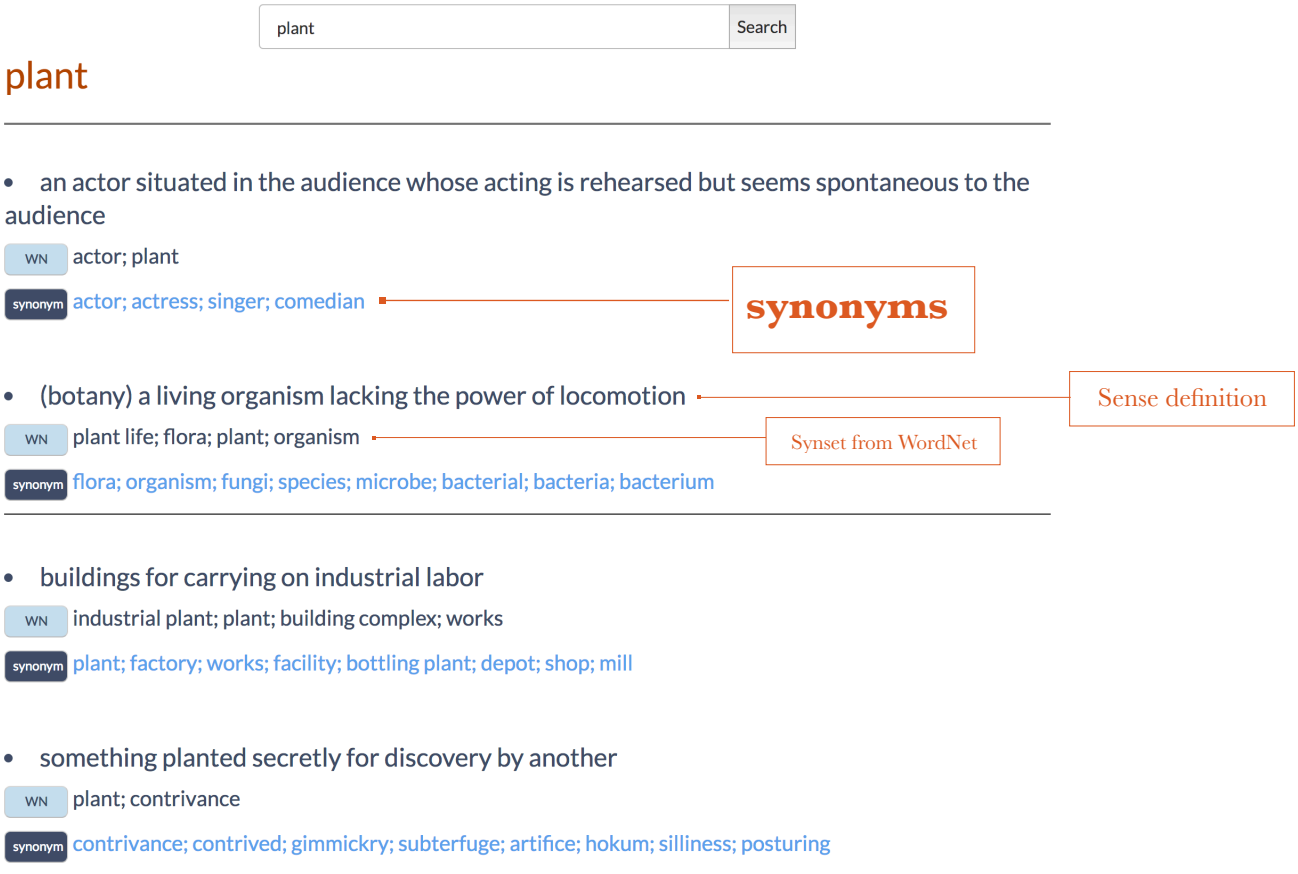
5. Take average of both vector of **positive_synset** and **negative_synset**.

Vector(Positive)	**v(plant_life) +v(flora)+v(plant)+v(organism) / 4**
Vector(Negative)	**v(plant) +v(actor)+v(contrivance) / 3**

6. Put these two vector into GoogleNews pertained word2vec model and get most similar synonyms.
7. Get the 200 most similar synonyms and set a filter that the synonym must has a similarity more than 0.2 with input word and exist in WordNet dictionary.
8. Sorted the rest of synonyms by multiplying its similarity with input word by standardizing term frequency.

Results

1. Enter website http://nlp-ultron.cs.nthu.edu.tw:3333
2. Input a word, for example, “plant”.
3. There are senses that “plant” contains.
4. There are sense definition and synset from WordNet of each sense.
5. The last one is the results of synonyms that using our method.



Reference

1. Ling, Wang, et al. "Two/too simple adaptations of word2vec for syntax problems." Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 2015.
2. Wolf, Lior, et al. "Joint word2vec Networks for Bilingual Semantic Representations." Int. J. Comput. Linguistics Appl. 5.1 (2014): 27-42.
3. Handler, Abram. "An empirical study of semantic similarity in WordNet and Word2Vec." (2014).
4. Wohlgenannt, Gerhard, and Filip Minic. "Using word2vec to Build a Simple Ontology Learning System." International Semantic Web Conference (Posters & Demos). 2016.
5. Fornander, Linnea, et al. "Generating Synonyms Using Word Vectors and an Easy-to-Read Corpus." (2016).
6. Qu, Meng, Xiang Ren, and Jiawei Han. "Automatic Synonym Discovery with Knowledge Bases." Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2017.
7. Hirao, Takuya, et al. "Vector Similarity of Related Words and Synonyms in the Japanese WordNet." Information Engineering Express 1.4 (2015): 21-31.
8. Yu, Mo, and Mark Dredze. "Improving lexical embeddings with semantic knowledge." Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). Vol. 2. 2014.
9. Foley, David, and Jugal Kalita. "Integrating wordnet for multiple sense embeddings in vector semantics." Proceedings of the 13th International Conference on Natural Language Processing. 2016.
10. Meyers, Adam. "Lexical Semantics 1: Word Senses and Word Similarity: WordNet and Vector Semantics."
11. Alistair Kennedy, Stan Szpakowicz (2014). Evaluation of Automatic Updates of Roget's Thesaurus. Journal of Language Modelling 2(1), 1-49.
12. Google Pre-trained word2vec. (https://github.com/mmihaltz/word2vec-GoogleNews-vectors/find/master)