LING 83600/CSCI 76000/ CSC 87100
Prof. Levitan
Yaffa Atkins
10-13-2022

HW3: Word Similarity

In this assignment you will evaluate several methods for computing word similarity using WordNet and word embeddings.

Motivation: Computing word similarity is a fundamental problem in NLP and used in many applications such as plagiarism detection, question answering, and historical linguistics.

Data: You are provided with the WordSimilarity-353 Test Collection, a subset of a total of 353 word pairs rated for similarity on a scale from 0-10, combined across annotators. The test set consists of 203 word pairs.

Task: You will implement two different approaches for computing similarity between words, and evaluate the approaches using the WordSim-353 dataset.

Method #1: WordNet Path-based Similarity
NLTK wordnet implementation: https://www.nltk.org/howto/wordnet.html
The starter code shows how to use the nltk wordnet interface.
Note: Wordnet similarity is defined for synsets, not words. You can choose the first sense for a given word, which represents the most common sense.
You can also experiment with a more sophisticated approach, which selects the pair of senses that gives the highest similarity score.

Method #2: Word embedding cosine similarity
We will use a gensim pre-trained models for this method:
https://radimrehurek.com/gensim/intro.html
The starter code shows how to load a pre-trained model and compute the cosine similarity between 2 words.

Evaluation:
For each method of measuring semantic similarity, you will compute two statistics for evaluation:
1. Correlation: the Spearman Correlation with human judgments of similarity.
You can use the Scipy implementation:
https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.spearmanr.html

2. Coverage: the percentage of word pairs in the database "supported" by the method (e.g. not all words are found in WordNet).

Similarity results table (3 pts):

Word1	Word2	WordNet path similarity	Word embedding cosine similarity
jaguar	cat	0.897	0.373
jaguar	car	0.308	0.475
king	queen	0.571	0.784
king	rook	0.381	0.254
tiger	zoo	0.533	0.367
tiger	cat	0.545	0.615

Similarity results question (1 pt): Reflect on the above table. Are the results what you expected? Why or why not?

I was surprised by some of the results on the above table. Jaguar and Cat have a very high WordNet path similarity which makes sense since a Jaguar IS A Cat. For this reason I was surprised they have such a low similarity score for cosine similarity, and that Tiger and Cat have pretty low WordNet path and cosine similarities. While a Jaguar is a type of car, their similarity is not very high since Jaguar is also often used in reference to the animal, still I would expect higher than a .308 from WordNet. Tiger and Zoo similarity does not surprise me since often tigers are not at zoos and zoos often are discussed with other animals.

King and Queen similarities did not surprise me since while they are quite similar, they are also opposites. King and Rook have lower similarities since their only connection is via chess but I would think their similarity would be higher since Rook is most commonly used when discussing chess which also discusses king.

Correlation with human judgments (across the entire 203 word pairs) (4 pts):

	WordNet Path Similarity	Word embedding cosine Similarity
Spearman correlation coefficient (to 4 decimals)	54%	56%
Coverage	100%	97.5%

Questions (2 pts):

1. Describe the WordNet Path Similarity approach. What are the advantages and disadvantages of this approach to measure similarity?

   WordNet Path Similarity is calculated using the Wu-Palmer similarity which looks at the depth of the two senses in the taxonomy tree and the depth of their least common ancestor node, calculating the length of the shortest path between the two synsets. It is calculated as the depth of the LCS (least common subsumer) divided by the sum of each word's depth. An advantage of this method is that it takes all synsets of the word into account, but this is also a disadvantage since it looks at the least common ancestor node. This means that words with real life high similarity in one meaning can have very low Wu-Palmer similarity if a different synset of the word and be given a very low similarity score.

2. Describe the word embeddings cosine similarity approach. What are the advantages and disadvantages of this approach to measure similarity?

   Cosine similarity looks at the cosine of the angle between the origin and the two vectors representing the two words, where similar words have similar coordinates. These word embeddings are generated using deep learning which analyzes large corpora to place the words on the grid, we used a pre-trained model for this assignment. Cosine similarity is calculated as the dot product of the vectors over the product of the length of the vectors. An advantage of cosine similarity is that similar words will be very close to each other and have very high similarity. A disadvantage is that since many words have multiple meanings and many different words have the same meaning, it is possible many words connected to the same word will be right next to each other and have high cosine similarity when they have little to nothing to do with each other.