Assignment 1

Deadline: Tuesday, 27th February (noon)

February 6, 2018

There are multiple ways to represent words. In this assignment, you are going to create multiple vector representations of words and compare the performance of these representations.

Question 1 (7pt):

Use the code provided in the accompanying jupyter notebook to load the "Alice in Wonderland" text document. You should not change the second cell of the notebook.

1. Implement the word-word co-occurrence matrix for Alice in Wonderland. Use a window size of 4 (*window_size_corpus* in the jupyter notebook).
2. Normalize the words such that every value lies within the range of 0 and 1.
3. Compute the cosine distance between the given words:
   - Alice
   - Dinah
   - Rabbit
4. List the 5 closest words to 'Alice'. Discuss the results.
5. Discuss what the main drawbacks are of a co-occurrence matrix solution.

Question 2 (13pt):

Build word embeddings with a Keras implementation where the embedding vector is of length 50, 150 and 300. Use the Alice in Wonderland text book for training. Use a window size of 2 to train the embeddings (*window_size* in the jupyter notebook).

1. Using the CBOW model.
2. Using the Skipgram model.
3. Add an extra hidden dense layer to the CBOW and Skipgram implementations. Choose an activation function for that layer and justify your answer.

4. Analyze the four different word embeddings:

   - Implement your own function to perform the analogy task with[1]. Do not use existing libraries for this task such as Gensim. Your function should be able to answer whether an analogy as in example 1 is true.

   $$\begin{gathered}\text{A king is to a queen as a man is to a woman} \\ v_{\text{king}} - v_{\text{queen}} + v_{\text{woman}} = v_{\text{man}}\end{gathered} \tag{1}$$

   - Compare the performance on the analogy task between the word embeddings that you have trained in questions 2.1, 2.2 and 2.3.
   - Visualize your results and interpret the results.

5. Use the word-word co-occurrence matrix from Question 1. Compare the performance on the analogy task with the performance of your trained word embeddings.

6. Discuss:

   - What are the main advantages of CBOW and Skipgram?
   - What is the advantage of negative sampling?
   - What are the main drawbacks of CBOW and Skipgram?

7. Load the pre-trained embedding on large corpora of GloVe (`http://nlp.stanford.edu/data/glove.6B.zip`) and Word2vec (`https://drive.google.com/file/d/0B7XkCwpI5KDYNlNUTTlSS21pQmM/edit`). You only have to use the word embeddings with an embedding size of 300.

   - Compare performance on the analogy task with your own trained embeddings from 'Alice in Wonderland'. You can limit yourself to the vocabulary of Alice in Wonderland. Visualize the pre-trained word embeddings and compare these with the results of your own trained word embeddings.

[1] Mikolov et al., (2013), Distributed Representation of Words and Phrases and their Compositionality. Check the introduction (`https://papers.nips.cc/paper/5021-distributed-representations-of-words-and-phrases-and-their-compositionality.pdf`)