Word embeddings - Practical exercises I

1. retrieve the word embeddings in word2vec format sample available on the github for this class : https://github.com/TimotheeMickus/lexical_resources.
   As a reminder, word2vec format starts with a line indicating V the number of word vectors and d the dimension of the vector space, followed by V lines, each representing a word vector, starting with a given word w, followed by d real numbers (float) representing the d components of the word vector $\vec{w}$.
2. write a function that computes and return a lookup dictionary and a matrix representing the word vectors based on the file.
   If the vector $\vec{cat}$ for the word "cat" is the 1st row of the matrix and the vector for "dog" stored at the 2nd, the dictionary lookup should look like this : {'cat': 0, 'dog':1} (index are 0-based in python)

Word embeddings - Practical exercises II

3. write a function that gets two words as strings of characters as parameters, and computes and returns the cosine of the two associated vectors.

   How do you deal with words for which you have no vectors?

4. return a function that takes a word as a string of characters and an integer k as parameters, and returns a dictionary of the k most similar words in the vector space, mapped to their scores.

   Use cosine similarity first, and then add a parameter to allow for other measures