



Lab 1 - Cosine Similarity & Accuracy: a Focus on the Analogy Task

Alberto Testoni, 9th November 2020



Nearest Neighbours with Cosine Similarity

We want to find the nearest neighbours of a word in a vector space.

What we need:

1. A matrix of all the word embeddings
2. A “dictionary” that maps each word to a row in the matrix, and vice versa
3. A distance function (cosine similarity)

Nearest Neighbours with Cosine Similarity

Length of the word embeddings

Vocabulary size (# words)

0.1	-0.3	0.2	...	0.1	0.6	0.8
0.2	0.4	0.1	...	0.2	0.5	0.3
...
-0.5	-0.8	0.4	...	-0.8	0.4	0.5
0.8	0.3	0.2	...	0.1	0.4	-0.9

word2idx

dog : 0
city : 1

....
friend : 3999
Paris : 4000

idx2word

0 : dog
1 : city

....
3999 : friend
4000 : Paris



Nearest Neighbours with Cosine Similarity

What is the word embedding of “city”?

0.1	-0.3	0.2	...	0.1	0.6	0.8
0.2	0.4	0.1	...	0.2	0.5	0.3
...
-0.5	-0.8	0.4	...	-0.8	0.4	0.5
0.8	0.3	0.2	...	0.1	0.4	-0.9

word2idx

dog: 0

city: 1

....

friend: 3999

Paris: 4000

idx2word

0: dog

1: city

....

3999: friend

4000: Paris



Nearest Neighbours with Cosine Similarity

Which word corresponds to the last row in the matrix?

0.1	-0.3	0.2	...	0.1	0.6	0.8
0.2	0.4	0.1	...	0.2	0.5	0.3
...
-0.5	-0.8	0.4	...	-0.8	0.4	0.5
0.8	0.3	0.2	...	0.1	0.4	-0.9

word2idx

dog: 0
city: 1

....

friend: 3999
Paris: 4000

idx2word

0: dog
1: city

....

3999: friend
4000: Paris



Let's Look at the Code!

How do we compute the nearest neighbours of a word in a vector space?

<https://colab.research.google.com/drive/1y9PtwOZ2E2k5aThj5cmVFPIDD24ZT-NI?usp=sharing>



The Analogy Task

- A proportional analogy holds between two word pairs:

$$x : y = a : b \text{ (} x \text{ is to } y \text{ as } a \text{ is to } b\text{)}$$

- For example:

$$\text{man} : \text{king} = \text{woman} : X$$

- An interesting property of word embeddings is that analogies can often be solved simply by adding/subtracting word embeddings.

$$W_{\text{king}} - W_{\text{man}} + W_{\text{woman}} \approx W_{\text{queen}}$$

nearest neighbour



Let's Look at the Code!

How do we solve an analogy with word embeddings?



Analogy Test Set (Mikolov et al., 2013)

- We will use the same dataset as in Baroni et al., 2014:
<http://www.fit.vutbr.cz/~imikolov/rnnlm/word-test.v1.txt>
(open the file and search for “:” to have a look at all the analogy types)
- We will evaluate the word embeddings using the accuracy metric:

$$\frac{\text{Number of correct predictions}}{\text{Total number of predictions}}$$



Let's Look at the Code!

How do we compute the accuracy of solving analogies in a test set?