CS224 - Assignment 5

O.L.

September 2019

1

1.a

embedding size used for character-level embeddings is typically lower than that used for word embeddings because number of words far larger then the number of characters and the different meaning of characters are fewer than the those of a words.

1.b

For char-based embedding:
For eq. (3) : $x_{emb} \in R^{m_{word} \times e_{char}}$
For eq. (4) : $W \in R^{f \times e_{char} \times k} \xrightarrow{f=e_{word}} W \in R^{e_{word} \times e_{char} \times k}$; $\quad b \in R^{e_{word}}$
For eq. (8) : $W_{proj} \in R^{e_{word} \times e_{word}}$; $\quad b \in R^{e_{word}}$
For eq. (9) : $W_{gate} \in R^{e_{word} \times e_{word}}$; $\quad b \in R^{e_{word}}$
So in total:

$$V_{char} \times e_{char} + e_{word} \times e_{char} \times k + e_{word} + 2 \times (e_{word} \times e_{word} + e_{word})$$

For word-based embedding:

$$V_{word} \times e_{word}$$

With the given numbers:

$$k = 5, V_{word} \approx 50,000, V_{char} = 96, e_{char} = 256, e_{word} = 50$$

char-based embedding:

$$96 \times 50 + 256 \times 50 \times 5 + 256 + 2 \times (256 \times 256 + 256)$$

For word-based embedding:

$$50,000 \times 256$$

Conclusion:

$$\frac{word - based}{char - based} = \frac{12800000}{200640} \approx 63.795$$

1.c

Convnet look at the all elements in the window simultaneously, so it doesn't suffer from decreasing the affect of the characters that were before the one that is now processed(compere to the sequential RNN). In addition it is faster because of the parallel computing as mention in the assignment.

1.d

1. Max-pooling:
   (a) Advantages;
      1. If there is a feature of interest - (like an eye in image to recognize whether it is a face), so if there is even a slightest signal we what to emplify it.
      2. amplify the signals.

   (b) Disadvantages:
      1. Ignoring the general trend of the data.
      2. Might make two features that behave the sa,me different e.g. for 2 featues with same center of gravity, but different variance to have different values.

2. Average-pooling:
   (a) Advantages;
      1. Getting the overall notion of the data - the center of gravity

   (b) Disadvantages:
      1. weaker signals.
      2. ignoring the variance of the data.

1.l

Corpus BLEU: 99.29792465574434

2

2.e

Corpus BLEU: 99.29792465574434

2.f

Corpus BLEU: 24.403835154375255

3

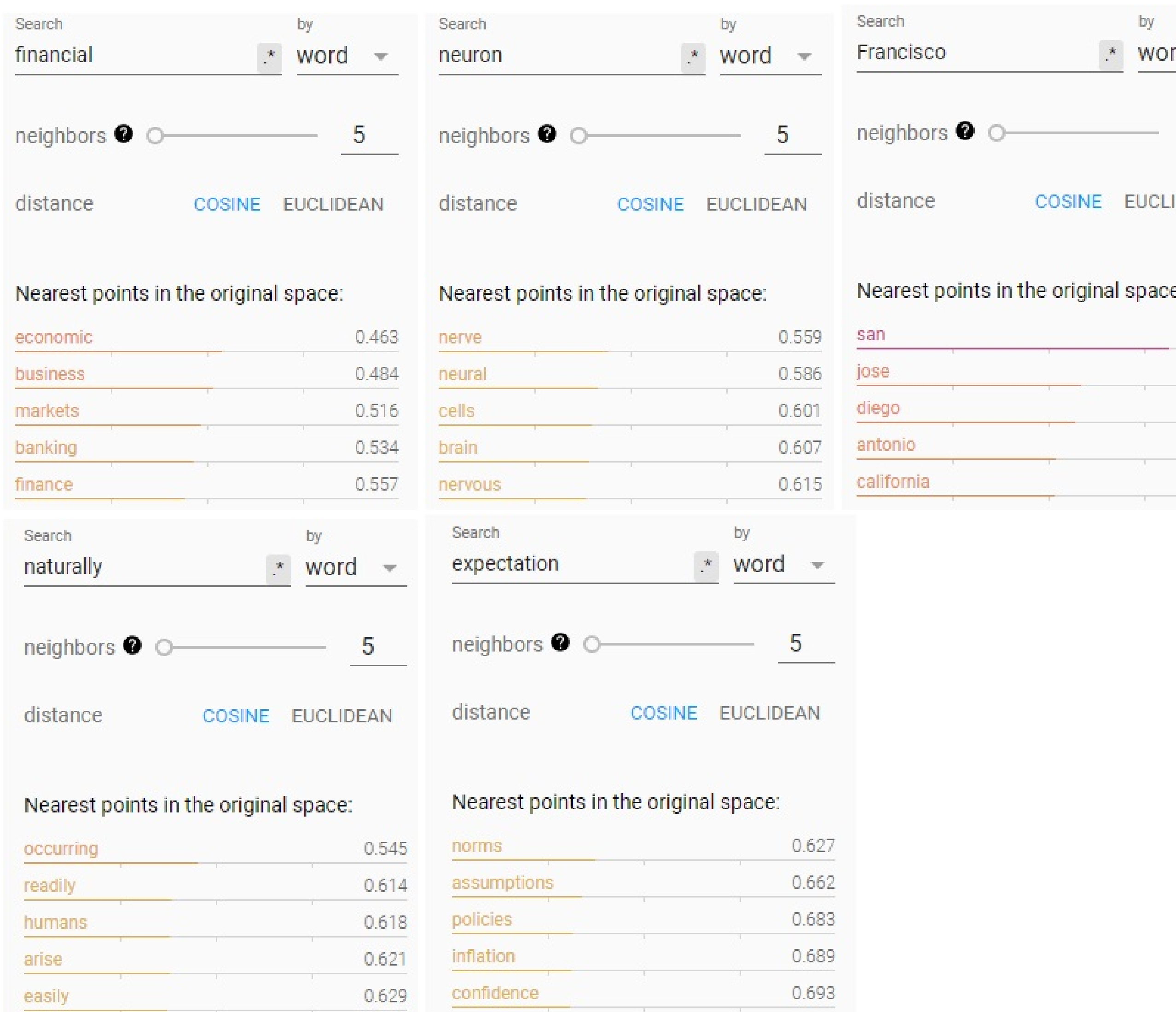
3.a

Run script "Question3_a.sh" in directory a5 to get:
traducir appear : 2
traduzco appear : 0
traduces appear : 0
traduce appear : 1
traduzca appear : 0
traduzcas appear : 0

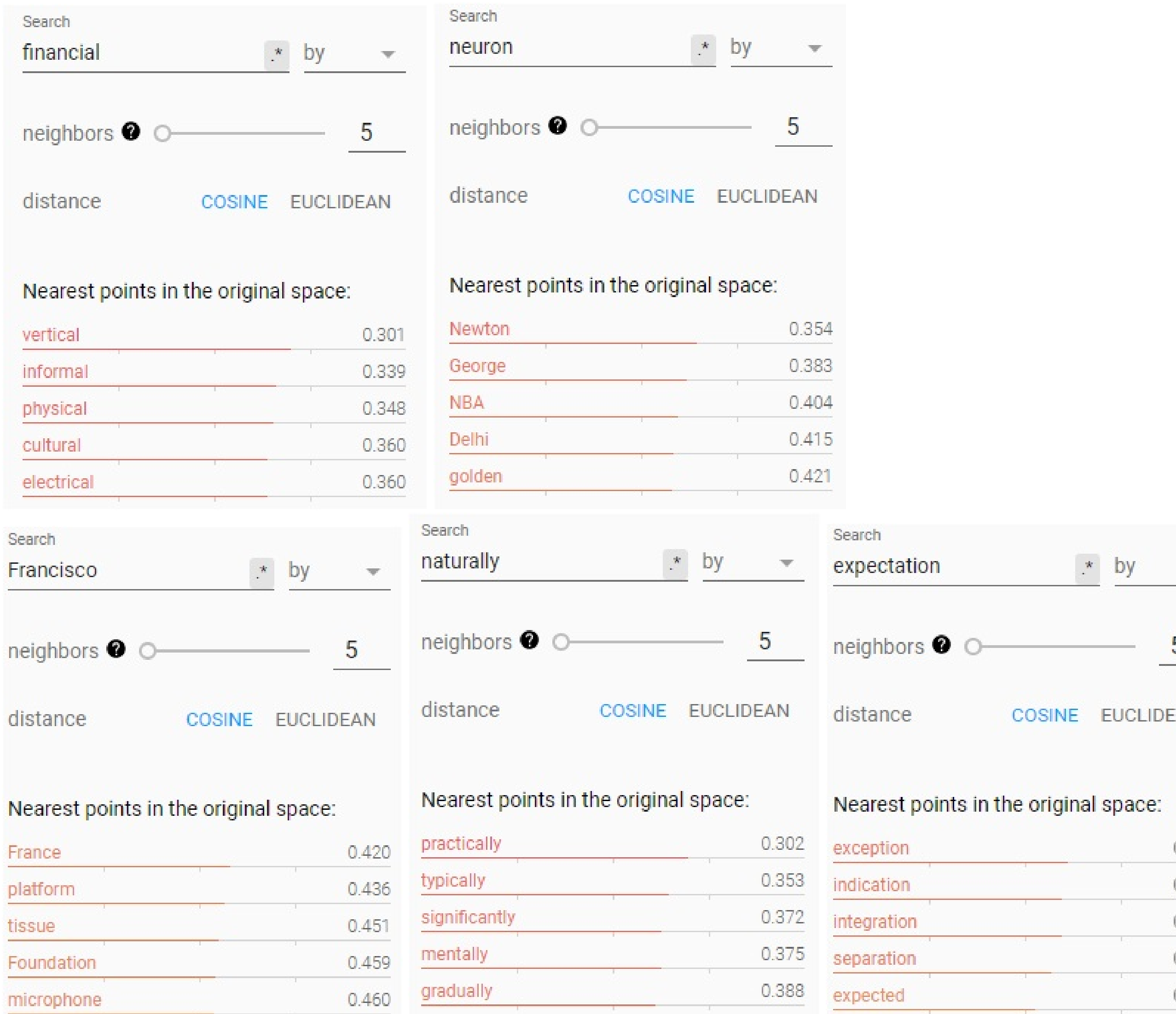
It is bad foe the NMT because the semantic meaning of the words is the same, so the model can insert any one of them in the correct place, furthermore the words are consider different, so it less likely to put the correct word meaning in the same context.
The new character-aware NMT model may overcome this problem, because it also get the syntactic meaning of the words.

3.b

3.b.1


Search
financial
by
word
neighbors
5
distance
COSINE
EUCLIDEAN
Nearest points in the original space:
economic 0.463
business 0.484
markets 0.516
banking 0.534
finance 0.557
Search
neuron
by
word
neighbors
5
distance
COSINE
EUCLIDEAN
Nearest points in the original space:
nerve 0.559
neural 0.586
cells 0.601
brain 0.607
nervous 0.615
Search
Francisco
by
neighbors
distance
COSINE
Nearest points in the original space
san
jose
diego
antonio
california
Search
naturally
by
word
neighbors
5
distance
COSINE
EUCLIDEAN
Nearest points in the original space:
occurring 0.545
readily 0.614
humans 0.618
arise 0.621
easily 0.629
Search
expectation
by
word
neighbors
5
distance
COSINE
EUCLIDEAN
Nearest points in the original space:
norms 0.627
assumptions 0.662
policies 0.683
inflation 0.689
confidence 0.693

3.b.2



3.b.3

Similarity modeled by Word2Vec:
    Semantic similarity
Similarity modeled by CharCNN:
    Syntactic similarity
Word2Vec is looking in the space for a similar word meaning (search by word context), while CharCNN search for the same structure of the word(trying to

predict the next character)

3.c