Second Year Project:
Natural Language Processing and Deep Learning
Week 4

Barbara Plank, Rob van der Goot

February 18, 2020

This is your assignment for week 4 of the course *Second Year Project*. You are expected to work on part I on Tuesday and part II on Friday.

After completing the whole assignment, you should:

- be able to implement a neural model to learn word embeddings, a simplified version of the continuous bag of words embeddings (CBOW) implementation of word2vec
- be able to obtain and analyze the word embeddings estimated from your CBOW model
- be familiar with HMMs and implement the most-likely-tag baseline and Viterbi decoding for POS tagging, explaining advantages and disadvantages of each

Requested reading: Chapters 7 and 8 from *Speech and Language Processing* by Jurafsky and Martin [1].

1 Part I: Learning word embeddings

Complete the exercises provided in the notebook: `learning_embeddings.ipynb`. They contain implementation exercises to create word embeddings.

2 Part II: Part II: HMM bigram POS tagger

In this exercise, we are going to create two POS taggers: one that is context-agnostic and one which takes context into account. For this exercise you can make use of the code in `hmm.ipynb`.

We will use the Danish part of the Universal Dependencies data, which is collected from https://github.com/UniversalDependencies/UD_Danish-DDT/tree/master. The data is in the following (tab-seperated) format:

```
# sent_id = dev-0
# text = Hvor kommer julemanden fra?
1  Hvor     hvor    ADV     _   _       2     advmod  _     _
2  kommer   komme   VERB    _   Mood=Ind|Tense=Pres  0     root  _     _
3  julemanden      julemand    NOUN   _     Definite=Def|Number=Sing   2    nsubj   _   _
4  fra      fra     ADP     _   AdpType=Prep 1       case  _     SpaceAfter=No
5  ?        ?       PUNCT   _   _       2     punct   _     _
```

We will only make use of the second and the fourth column, i.e. the words and the UPOS tags. The `read_conll_file` function in `myutils.py` reads these files and returns a list of pairs of sentences and tags. For the previous example this would be:

```
[(['Hvor', 'kommer', 'julemanden', 'fra', '?'],
  ['ADV', 'VERB', 'NOUN', 'ADP', 'PUNCT'])]
```

More information about the tag-set can be found on: https://universaldependencies.org/u/pos/all.html.

You are provided with an HMM class, which already estimates the emission as well as the transition probabilities. The class and the rest of the assignment can be found in: `hmms.ipynb`.

References

[1] Jurfasky and Martin, In Preparation. *Speech and Language Processing (3rd ed. draft)*. Available at https://web.stanford.edu/~jurafsky/slp3/