Assignment 2: Neural sequence learning

What we saw this was a lot of structures for sequence learning in neural nets, and also some application of these structures. E.g. PoS tagging, word embeddings, machine translation, dialogue.

For this assignment, implement one of two styles of sequence to sequence learning.

Option 1: A tagger

Implement, using LSTM and GRU (or any other RNN setup), a tagger for part-of-speech tagging or named entity recognition. You can choose any dataset or language you like, but I recommend the Universal Dependencies data, English or Danish, for part of speech.
www.universaldependencies.org

Implement the tagger in any framework of your choice.

Evaluation: for part-of-speech, measure:
- token accuracy %, i.e. how many tokens get the right label;
- how many sentences were labelled completely correctly (e.g. sentence accuracy %).

Try also running the system for German. Compare LSTM with the other RNN type. Which does better? Why?

The analysis

What did the system do well? What kind of mistakes did it make? Try to tune the learning rate, epoch count, and batch size (these are all *hyperparameters*). Which work best?

Find what the most common errors were, and give some examples of them - and examples of the tagger working correctly.

Assignment hand-in

Essay
Description of what you did and why, describing your general code, and the answers to all of the above questions in the work description. About 1000 words.

Code
Include your code. A link to a Colab notebook is best. Test the whole notebook first. I will run it myself using “reset and run all cells”.

How?
By LearnIT.

When?
July 29th. Good luck, and I hope you enjoy it!