LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

Erweiterungsmodul: Machine Translation **SS 2023**

Exercise Sheet

Basic Translation Models in Google Colab

This year, please do this assignment *over the summer break*, after the exam! You do not need to submit it. The first two tutorials are optional, but they will help you with your understanding of the required third tutorial and are strongly recommended.

The first two tutorials should work fine on your laptop, but Google Colab is recommended. You will probably need to run the third tutorial on Google Colab for it to train in a reasonable amount of time (30 minutes or so), but you could still try it on your local CPU or GPU first.

Exercise 1. OPTIONAL: Do the PyTorch tutorial "NLP From Scratch: Classifying Names with a Character-Level RNN" by Sean Robertson.
`https://pytorch.org/tutorials/intermediate/char_rnn_classification_tutorial.html`
Report the final exact performance on English and German.
What is a simple way to improve the German results, and what performance do you get with this improvement?

Exercise 2. OPTIONAL: Do the PyTorch tutorial "NLP From Scratch: Generating Names with a Character-Level RNN" by Sean Robertson, note that you need to have done the previous tutorial to do this one.
`https://pytorch.org/tutorials/intermediate/char_rnn_generation_tutorial.html`
The implementation of this network is counter-intuitive, in that it requires a start letter. For instance, the three Russian samples are generated using the letter specified in the second parameter of the call to the "samples" function, i.e., using the three letters R,U,S, to get three probable words starting with R, U and S respectively.
Modify this network to randomly select the next letter (using the posterior probability of each letter ranging from 0 to 1, and summing to one) instead of always taking the most probable letter. Discuss your modifications to achieve this. Show sample outputs.

Exercise 3. Do the PyTorch tutorial "NLP From Scratch: Translation with a Sequence to Sequence Network and Attention" by Sean Robertson, but translating from German to English (rather than French to English as in the tutorial).
`https://pytorch.org/tutorials/intermediate/seq2seq_translation_tutorial.html`
Decode some test data for which you have references, and compute BLEU.
Then visualize the attention at three different stages of training, and discuss how the attention is changing at these different stages over time. Discuss the relationship of the attention activations you observe with a diagonal attention activation.