Final Task Info Sheet

Implementing ANNs with TensorFlow

October 2018

1 Final Task

This course will not end with a final exam like most courses do. But because the focus of this course is on the implementational side you will work on a project at the end of the semester. You will work in the same group in which you work on the homework.
There is no official starting date, but you have to decide on a topic until 14th of January. Please write me an E-mail if you decided on a topic.
The submisssion deadline is 23.59 on 28th of February.

2 Topic

There are a lot of possibilities what you could work on for the final task, but they can be divided into two categories:

2.1 Paper

Find an ANN paper that interests you and reproduce the results. If the dataset that is used in the paper is too large or not available it is totally okay to use some small toy dataset. In the end it is not about the results but about the code you produce.

2.2 Own Idea

Come up with your very own idea of an architecture or of an experiment and implement it. Here again it is not important that your architecture gives astonishing results or your experiments work out as expected.

At some point I will upload a presentation or list with suggestions. If you find a paper your own or you really come up with your own idea please talk to me or write me an E-Mail so we can design an appropriate task together.

3 Report

Additionally to the code you have to write a small report of 5-6 pages. Please stick to this length. The report should include

- introduction into your task
- theoretical knowledge necessary to understand your model or experiment
- description of your model or your experiment
- evaluation of the results

Overall a reader should have an easy time to dive into your code after reading your report.

4 Grading

As it is very open how exactly your task will look like I find it hard to design a clear outline of how I will grade your task. Thus I will just give the aspects that I will assess:

4.1 Code

- structure of your project (file naming, file organization)
- structure of the code (functions for repetitive parts, use of built-in functions where available, readability)
- comments (code should include comments where they are needed)
- executability of code (does it run?)
- performance (did you make an effort to achieve good results?)
- complexity of the task (will not heavily influence, but will be taken into account)

4.2 Report

- structure of the report (readability)
- visualization (plots and figures)
- content (see chapter 3)
- helpfulness (does it help in understanding your project and your code?)