

Exercise 2 Report

First of all, implementing the network took around 15 hours, because I have never used tensorflow before. In the beginning, the way operations should be defined, how the graph should be structured, how to save and restore it all seemed unintuitive to me, so it took quite some time until I was familiar with it. Secondly, running the implementation very often resulted in memory overflow (`std::bad_alloc`), even if I use one of the provided pool computers. Because of this, the script would raise error after 2 hours of training and would not save anything. I am still not sure if there is a problem with my implementation or it is just that the memory of the pool computers are very limited. I could test the influence of learning rates on the learning curve, but I could only test filter sizes of 1 and 3, but not 5 and 7. I also could run the random search for the same reason, although I still wrote the code for running and plotting it.

As for different learning rates, the learning rate of 0.1 gave the best result, with the final test accuracy of ~ 0.9892 . Other learning rates also work, but it seems it just takes longer for them to converge. In particular, I would not recommend to use the learning rate of $1e-4$ as it is unnecessarily slow.

Between the filter size of 1 and 3 which I could run, using filter size of 3 gave consistently better results at every epochs. Using filter size of 1 therefore seems suboptimal, but its learning curve still suggests that it converges to somewhere around $0.9 \sim 0.95$, which is reasonable.

Overall, this exercise was frustrating because of the memory error and wasting a lot of time rerunning the experiment over and over, while not knowing if it is problem on my part or not.