

Week 4 Report

2023-12675 박지호

In this week's semina, Mathhias Gallé introduced us to the world of synthetic data and code. Even before the seminar started, I found this topic to be the most interesting of all the fields we covered thus far. As much as I found the specific optimization techniques introduced in previous lectures interesting, I thought that this seminar would give me useful insights on the AI industry as whole.

The seminar started off with a graph showing that more and more companies are starting to utilize AI in some way, shape, or form. The presenter claimed that we should see AI as a new way to increase productivity, as a new way to browse the information we need, like search engines such as Google. I was skeptical about using LLMs for research before, as search engines usually provide more immediate result without a risk of hallucination. To this day, I still believe that we shouldn't use LLMs as a primary source of information. However, recently, I found out how useful of a tool it can be, if only I cross reference it with a valid source, and finding a valid source is much easier when you have something to work from. Hence, I now admit that I simply didn't understand how to use LLMs properly before, and I completely agree with the presenter's take on the LLM and productivity.

Then, the presenter explained the basic premise of the LLM's training method, like back propagation, and the seq2seq. These breakthroughs lead us to a model that has output data space equal to the input data space. Therefore, we can theoretically use the output data as train data again. The presenter then listed few reasons people find this idea attractive. Not only are we running out of data to train from, but the legal issue revolving around the data usage also further limits the number of available data even. Also, it is very difficult to distinguish between human and machine generated text. However, the presenter then showed a research showing that performing pure self-training recursively could cause a model collapse, where the data distribution no longer accurately reflects the data distribution of the real world.

The presenter then told us of ways to somewhat rectify this issue: using a model known to perform better than the model we wish to train, using a constitutional AI, where we ask the AI to rewrite the existing data with certain constitutions in place, and lastly, using the reward model to filter out good results and only employing those for further training. I am still not completely convinced that these measures necessarily ensure the quality of the generated data. However, the fact that these methods are being actively used in the industry should serve as enough proof.

The seminar then moved on to the next topic: code. The presenter described code as the 'formal language', and introduced us to the concept of test driven development. In the context of code generation, synthetic data creation receives prompt as an input, and generates the code and the unit test. The presenter then shared with us the result of a research they conducted. In the research, they trained two models, one with purely synthetic data, and the other with synthetic data and human annotated data, to test the viability of synthetic data. However, it turned out that with enough synthetic data, the first model, the one without any human annotated data, performed better than the model with it. This result proves that not only is the synthetic data viable, it is more viable than the human annotated data itself. Obviously, I found this very interesting, even considering the formal nature of code that the presenter mentioned. The presenter mentioend that they still use human annotations for training in certain scenarios like complex problems, problems which cannot be unit tested, or programming language with extremely low resources.

I had something scheduled for the time of the seminar, so I had to watch the recorded version of the lecture and therefore couldn't ask any questions. However, if I could, I wanted to ask why the presenter thinks the human annotations detriments the model performance, contrary to our intuition.