

Our project is to train an AI model using Retrieval-Augmented Generation to produce statics problems for the ENED teachers to use. These have to generate various scenarios that can introduce the same numbers but in contexts which are relevant to whatever degree the student is going into. It also needs to generate unique numbers that, when taken together, produce a statics problem that is solvable and easy enough for students to work through. This will certainly require training an existing Large Language Model that already exists on the market, because none of us have the experience to produce a sufficient neural network to match that performance. As for the number selection, I figure that since these are essentially systems of equations we should be able to start in row echelon form and step backwards into a question that isn't just impossible.

In terms of experience with AI, I took the AI Principles and Applications (CS 4033) and Machine Learning (CS 5137) classes with Michael and JP. Of those it's really only Machine Learning where we learned about working with neural networks such as Large Language Models. My memory is a little bit spotty, but I expect I should be able to refresh my memory as we go along. We also potentially talked about RAG too, but I don't remember that discussion in class because it wasn't on the exam. If working backwards from row echelon form does end up being the right call then we all also have experience in the required Linear Algebra (MATH 2076) class to have experience with matrices.

My Co-op experience was only ever as a software development intern for Siemens. Nothing I did with Siemens really applies to anything we did, I didn't do much work with AI. The only AI project I worked on was using an AI to do unit testing using image recognition, so the best thing I can contribute is maybe putting together an automated UI test. Then, this isn't such a big project that automation would really be a worthwhile investment. My best skills from

Siemens is just being able to work as a software engineer and work on whatever project I'm put on. And maybe some more Git exposure than the other?

The motivation for the project came from JP. He was talking with the ENED team and they suggested the need for an automated way to put practice problems together for their students. This was great because Michael is passionate about working with machine learning, so JP offered to make it if the ENED teacher was willing to advise us. I personally had no reason to reject the idea from JP, seeing as all three of us had worked on AI projects before, so I agreed. I think that having worked on similar projects before means we have a smaller struggle getting used to the team dynamic, since it is in that same field.

As before, our plan mainly revolves around RAG. Since we'd be using an off-the-shelf LLM, having a way to direct the machine to specifically statics problems means that the accuracy of the question is greatly improved. What's more, RAG is an easy enough feature to implement for all of us to improve the result without needing advanced machine learning. To ensure that the resulting numbers are solvable they could also be first put in row echelon form and then scrambled using row manipulations to ensure that not only is the example reasonable, but the numbers are enough to solve the question. Nothing here should be new, it's all just about polishing on what we know already. The result should be a machine that can generate valid question numbers, then produce a simplified statics context to make into a usable word problem in class.