CSE 584 Final

Maxwell Dailey

December 2024

1 Introduction

In this document, I will present several potential research questions that the dataset I created could be used to help answer. I will also outline the experiments I would perform to answer each question.

2 Question 1

Question: In my dataset, there are many instances where the LLM responds incorrectly because it fails to realize that some part of the premise of the question is invalid. If a model was fine-tuned to spot these faults (without any additional prompts being added to the questions), would this lead to improvement in its ability to answer questions because now it is putting more consideration into each part of the question?

Experiment: First, I would choose a Question-Answer dataset (of which there are many), and then I would evaluate the performance of ChatGPT-4o-mini on that dataset to get a baseline. Then, I would find another Question-Answer dataset and combine it with the one I created so that the new resulting dataset is one part of each. Using this newly combined dataset, I would fine-tune ChatGPT-4o-mini with the goal of having it learn to, based on its knowledge, identify questions that have partially or completely faulty/impossible premises. After this, the fine-tuned model would be re-evaluated on the initial Question-Answer dataset to determine what effect, if any, teaching the model to identify faulty questions had on its ability to answer questions correctly.

3 Question 2

Question: Part of the instructions for generating the dataset was to not prompt the LLM that some part of the premise of the question could be faulty, as doing so would almost always lead it to discover any faults to the best of its knowledge. This would be great for finding impossible questions, but those account for a very, very small amount of the questions people ask ChatGPT-4o-mini, so

this particular prompting would be nearly useless. What would be much more useful is to investigate: Without explicitly saying that the question could be faulty, is there prompt engineering that can be added to an existing question that would cause ChatGPT-4o-mini to "think" about the question more deeply to be better about identifying faulty questions and potentially provide a more insightful answer to valid ones?

Experiment: I will first brain-storm a number of potential prompt additions (excluding simply telling ChatGPT-4o-mini, in one way or another, that the questions could be flawed, of course) that I think would improve the depth of the answers given for questions and would allow it to spot flawed questions. I would also select an LLM Judge that I believe would be capable of best judging an answer's in-depthness. Then, I would find a Question-Answer dataset and combine it with the one I created so that the new resulting dataset is one part of each, and evaluate ChatGPT-4o-mini on the combined dataset to get a baseline performance. This performance will be measured using a combination of how many of the faulty questions ChatGPT-4o-mini identified and how in-depth the LLM Judge considered its answers to the valid questions to be. Next, I would repeat this evaluation process for each of the potential prompt additions. Lastly, I would compare and analyze the results and, based on these results, will try to come up with more prompt additions that would work even better.

4 Question 3

Question: In my dataset, a question is made invalid in two ways. The first is through a statement of fact or description of events that are impossible that indirectly relate to the question asked, and the second is through a statement that more directly relates to the question being asked. I think an interesting question to answer is: Which way is easier for an LLM to identify and explain? I feel like it should be the statements that relate more to the questions, as they are closer to the overall "train of thought" of the LLM but the only way to know for sure is to run an experiment.

Experiment: As it stands now, ChatGPT-4o-mini doesn't identify any invalid aspects of the questions in my dataset, so the questions need to be changed slightly in order to hopefully discern a difference in a model's ability to identify the invalid parts. I propose adding the sentence, "If part of the premise of this question is invalid, say so and identify the part and explain why." As I mentioned previously, explicitly telling ChatGPT-4o-mini to be aware of invalid questions means that it will almost always recognize one, but that doesn't mean that it will necessarily correctly identify the part or correctly explain why that part makes the question invalid. Luckily, the invalid part of a question in my dataset is usually the second and/or third sentence, so one can easily be parsed out and known. This is important because it would be impossible to know if the LLM correctly identified the invalid part if we didn't have a way of sectioning

out the invalid part ourselves. So, for the experiment, we will first add the previously mentioned sentence to each of the questions in the dataset and add a new column that will contain the part that makes each question invalid. Then, using this modified dataset, we will use an LLM Judge to evaluate how well the model identified the invalid part of each question and how well it explained why it was invalid. Given this information, we will compute a value that represents the performance on both types of questions and can then compare and analyze any difference seen.

5 Question 4

Question: Each of the previous questions have only made use of the question part of my dataset and not the answer part. As such, another interesting question is: Prompted that something is wrong, given a question and answer pair as input, will an LLM reach the conclusion that the question is invalid?

Experiment: Compared to the others, this experiment is relatively simple. First we will create a new dataset with only inputs, as the output for every question should be that the answer is wrong because the question is invalid. The inputs in this dataset will be created by taking the question and answer pairs from my dataset and adding them as context after the prompt: "There is an issue with the provided answer to the provided question below; what is it?" The LLM will be run on this dataset, and its outputs will be fed into an LLM Judge to determine how similar the LLM's output is in meaning to a sentence like "Part of the premise of the question is invalid/impossible, so giving any answer outside of stating that will be wrong." By compiling these similarity-in-meaning scores we can determine how well a model can re-evaluate its answer to a question (and re-evaluate the question itself) given that it is told something is wrong.