Final Project Report

Asfahan Shah (aks7824@psu.edu)

1 Research Question and Motivations:

Large Language Models (LLMs) have become integral to our daily lives, with startups like Perplexity using them to replace traditional information retrieval methods, such as web search. This trend opens exciting prospects for the future, but also raises critical questions: Are LLMs robust enough for this role? Specifically, there are two potential issues to explore:

- LLMs may lack adequate domain-specific knowledge.
- The internal workings of LLMs may misinterpret or overlook crucial aspects of a query.

Both of these issues are important and need to be studied to get a full landscape of abilities of LLMs. In this project, we will try to touch upon these issues by conducting a study on popular LLM model ChatGPT 4.0 mini. To conduct this study, we will create a dataset of faulty science questions (with various error types detailed in the dataset section). By analyzing the failure of LLM on this dataset, we aim to answer the following research questions:

1. In which domains and on which types of errors do LLMs fail?
2. What are the underlying reasons for these failures, and do they stem from LLMs' lack of domain-specific knowledge?
3. What potential methods could help overcome these failures in LLMs?

In this context, "failure" is defined as the inability of an LLM to detect a fault in a question and its attempt to answer the question as if question were correct.

2 Dataset:

To address this issue, we have developed a dataset of 70 faulty science questions, equally divided across 7 domains: Physics, Biology, Astronomy, Meteorology, Engineering, Robotics, and Mathematics. The questions contain various types of faults, including:

1. Factual inaccuracies (specific to the domain, not common knowledge or universally known facts).
2. Mathematically plausible but scientifically impossible scenarios.
3. Oversimplification of complex phenomena.

These errors are crafted to avoid manipulating universally accepted facts (e.g., the speed of light is 3 m/s) and instead focus on domain-specific inaccuracies (e.g., Mercury's temperature is 1000°C). These inconsistencies may not be immediately obvious, which is critical for evaluating the LLM's ability to detect them.

3 Testing

We tested the 70 questions using the ChatGPT 4.0 mini model. Each question was tested in isolation in a private browsing window to ensure that previous questions did not influence the model's responses. After receiving an answer for a given question, a new private window was opened to test the next question.

The results were consistent across all 70 questions: the LLM failed to detect the faults and provided standard solutions without addressing the errors.

Interestingly, we also tested whether the LLM possessed domain-specific knowledge. For example, instead of asking the faulty questions like "Mercury's daytime temperature reaches 1000°C, while nighttime is -180°C. If a 1 kg aluminum block is placed on Mercury's surface, what will be its volume at noon?" directly, we asked the model to verify the validity of the statement by prompting "Is the following statement valid or not, Mercury's daytime temperature reaches 1000°C, while nighttime is -180°C." The LLM correctly identified the statement as false and provided a justification. This suggests that the LLM does have domain-specific knowledge but struggles to apply it in the context of faulty questions.

3.1 Discussions

LLMs failed across all domains (Physics, Biology, Astronomy, Meteorology, Engineering, Robotics, Mathematics) and all types of errors, including factual inaccuracies, mathematically plausible but scientifically impossible scenarios, and oversimplifications.

Though we note that LLM was able to detect faults in questions that violated universally accepted facts, such as "1=0" or "the speed of light is less than sound." In these cases, the model correctly identified the errors and provided justifications. However, when the fault was more subtle and domain-specific (e.g., "Mercury's daytime temperature reaches 1000°C"), the LLM was unable to detect the issue and simply solved the problem as if it were correct. But if we rephrased the question to verify the validity of statement rather than solve the question, LLM was able to correctly identify these subtle and domain-specific errors.

This suggests, the failure is not due to a lack of domain-specific knowledge. Instead, it seems to stem from the LLM's inability to focus on identifying faulty domain specific subtle facts in contexts of solving questions. The model tends to prioritize solving the question rather than validating its premises. This behavior mirrors how a non-expert human might approach a domain problem—focusing on solving the problem without considering subtle errors in the question itself.

4 Solution

The cause of failure seems to be a lack of focus on the faulty parts of a question. To address this, we propose a dual-prompt approach. The idea is as follows:

1. The LLM receives the initial faulty question (e.g., "Mercury's daytime temperature reaches 1000°C, while nighttime is -180°C. If a 1 kg aluminum block is placed on Mercury's surface, what will be its volume at noon?").
2. Instead of solving it immediately, the LLM first extracts the premise part of the question (e.g., "Mercury's daytime temperature reaches 1000°C, while nighttime is -180°C") and appends it to a new prompt: "State whether the following statement is valid or not."
3. The LLM evaluates the truth of the extracted statement. If it is found to be false, the original question will not be solved. If the statement is true, only then will the model proceed to solve the problem.

This dual-prompt mechanism acts as a safeguard to ensure that the LLM validates the premise of a question before attempting to answer it. The main drawback is the increased computational time and cost due to executing two prompts.

5 Conclusion

In conclusion, we have shown that LLMs fail to detect faulty questions across various domains due to a lack of focus on the question's premise, despite having domain-specific knowledge. A dual-prompt approach could mitigate this issue by ensuring that the LLM first validates the question before proceeding with the solution, though this comes at the cost of additional computation.