Evaluating LLM Performance on Faulty Science Questions

Lakshmi Chandrika Yarlagadda (lvy5215)

Introduction:

This project deals with the evaluation of the performance of large language models in answering faulty science questions. The aim is to determine whether LLMs are able to identify and handle logical and factual inaccuracies in questions about various scientific domains. The method entails creating a dataset of faulty science questions that were put through models like ChatGPT, GPT-4, Claude-3-Opus, Qwen, and Llama. Key findings indicate that LLMs are more likely to fail when the errors are subtly hidden in the question and are in logical domains such as math and physics but perform better on factual subjects like biology and history. Subsequently, prompts for reasoning and clarification may help LLMs to realize and correct errors more effectively.

Dataset Curation:

I developed a dataset of 125 questions, which were then assessed through GPT and Gemini. This yielded a total of 250 questions. I have tested around 1000 questions and curated this dataset of 250. The questions cover some of the disciplines in sciences. Precisely, I included open-ended questions in Mathematics and Physics, as well as Chemistry questions related to the properties and structures of materials.

I created MCQs in both Geology and Biology-which includes Botany and Zoology. To frame the MCQs I used questions from the JEE Exam because of their comparatively higher level of difficulty. I asked GPT to change the correct answers to their nearest incorrect options. This made the questions subtly wrong while remaining inconspicuous.

Regarding the open-ended questions, I asked GPT to alter some JEE physics questions slightly to conceal a small mistake or to challenge the known laws of physics. I tested the dataset on four different models: ChatGPT, Gemini, Qwen, and Llama. Models Qwen and Llama performed better in error detection within the questions and were, therefore, excluded from further testing. ChatGPT and Gemini performed less well when it came to error identification, specifically when the errors were subtle.

Research Questions:

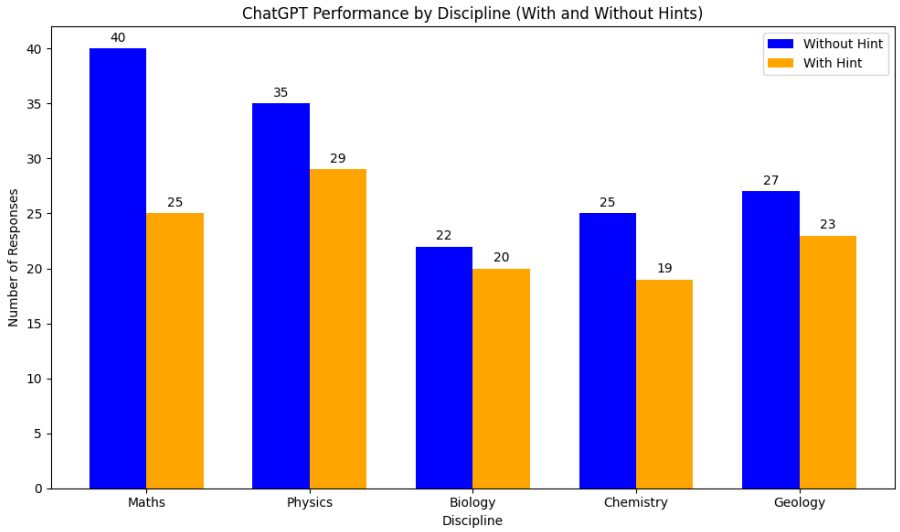
1. How does the LLM perform when external hints are provided that the question might be faulty?

Experiment:
To test how LLMs perform when given external hints, I modified the way I prompted the questions. Instead of directly providing the questions, I added hints indicating that the question might be faulty. This approach aims to see if the LLM can better identify the issues when it is explicitly told to look for faults.

Prompt:
For example, I used the prompt: *"Point out the faults in the question if any and answer the question:*

'If a triangle has angles 90°, 60°, and 70°, and two sides are 5 and 7, what is the third side?'" I then followed up with a check: *"Check if this is valid."*

Observation:
When I provided these extra hints, the LLMs were able to identify the faults more easily compared to when I simply asked the question without any hints. For instance, in the case of the triangle question, the LLMs could recognize that the sum of the angles (90° + 60° + 70°) is 220°, which is not possible for a triangle, and therefore the question is invalid. When the LLMs were not given these hints, they focused on solving the problem as if the question was valid, often ignoring the fact that the angles did not add up correctly.



I have given 50 questions in each domain to ChatGPT, and the same questions were given again with hints. The number of responses after hints were reduced as the LLM can identify the faults because of the hints.

2. How does the LLM respond incorrectly to obvious/hypothetical questions versus subtly faulty questions?

Experiment:
To investigate how LLMs respond to obvious faulty questions versus subtly faulty ones, I tested the models with two versions of the same type of question: one where the fault was obvious and another where the fault was hidden within the question. The goal was to see if the LLM could detect clear issues or if it would focus on solving the question when the error was more subtle.

Example:

- *Obvious Faulty Question*: "The angles of a triangle are 90°, 60°, and 70°. Check if it is a right-angle triangle?"

The LLM correctly responds that the triangle is not possible because the angles add up to more than 180°.

- *Subtly Faulty Question*: "If a triangle has angles 90°, 60°, and 70°, and two sides are 5 and 7, what is the length of the third side?"

In this case, the LLM ignores the fact that the angles are invalid and proceeds to attempt solving the problem, giving an answer based on the faulty premises.

Observation:
When I tested obvious, hypothetical, and nonsensical questions, the LLMs were usually able to recognize these scenarios as unrealistic or hypothetical. For example, when the question clearly stated an impossible situation (like the sum of angles in a triangle being more than 180°), the LLM recognized the error. However, when the fault was made subtle such as by hiding the incorrect fact within a more complex question the LLMs tended to focus more on solving the problem at hand rather than identifying the error. This suggests that when the model is distracted by other parts of the question or when the fault is not immediately obvious, it might fail to notice the underlying mistake and still attempt to answer the question based on incorrect assumptions.

3. How do the LLMs respond to ambiguous questions?

Experiment**:**
When testing how LLMs handle ambiguous questions, I provided intentionally unclear or vague questions to see how they would respond. I wanted to check if the LLM would seek clarification or if it would just make assumptions and give an answer.

Example**:**

- *Question*: "What is twice of thirteen added two?"
  This question has two possible interpretations:
    1. Twice of thirteen, to which two is added: 2×13+2=282 \times 13 + 2 = 28.
    2. Thirteen added to two, which is then doubled: 2×(13+2)=302 \times (13 + 2) = 30.

Observation**:**
Most LLMs defaulted to one interpretation of the question and gave an answer based on that, even though multiple interpretations were possible. This shows that LLMs tend to make assumptions rather than seek clarification when dealing with ambiguous questions.

4. Is it easy to trick the LLM with logical or factual faulty questions?

Experiment:
While creating the dataset, I tested the LLM with both logical and factual faulty questions to see how easily it could be tricked. I included subtle inconsistencies or incorrect facts in some of the logical or numerical questions to observe how the model would handle them. I also included factual questions to see if the LLM could identify inaccuracies more easily.

Example:

- *Logical Question with Fault*: "If a triangle has angles 90°, 60°, and 70°, and two sides are 5 and 7, what is the length of the third side?"

The LLM may ignore the fact that the angles add up to more than 180° and attempt to solve the problem, providing an incorrect answer.

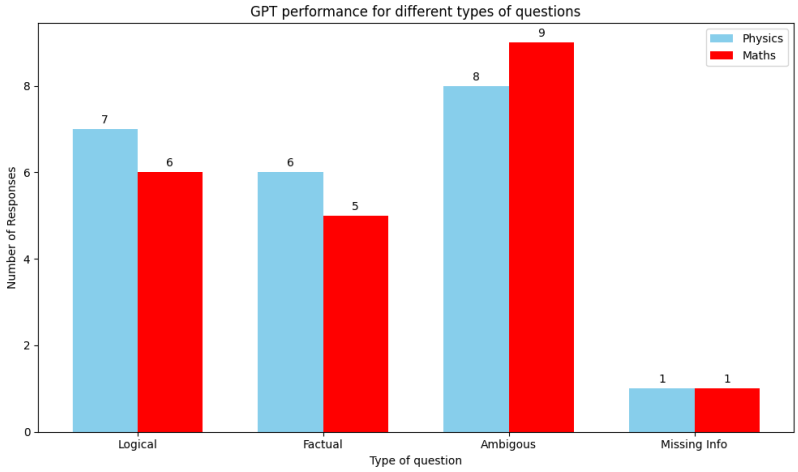
- *Factual Question with Fault*: "Who discovered gravity in 1880?"

The LLM might ignore that gravity was discovered in 1600s and answer the question as Isaac Newton.

- *Missing Information*: "A car accelerates uniformly at 5 m/s^2. How far does it travel in 10 seconds?" – velocity not provided

Observation:

I found that it is hard for the LLM to find faults with numerical or logical questions. When the logical structure of a question is altered slightly, the LLM often attempts to solve the question without recognizing the underlying flaw. On the other hand, when the questions are factual, the LLM is usually able to identify the inaccuracies and provide the correct information. This suggests that the LLM handles factual errors better than logical or numerical inconsistencies. But when missing information questions are asked the LLM is able to recognize the missing information but sometimes it defaults and assumes something and proceeds with the solution.



10 questions were given for each category and the above are the statistics for how many questions ChatGPT gave a response without finding faults. Most of the logical questions are answered without finding faults. All the ambiguous questions are defaulted to something. A couple of factual faulty questions are answered. Most of the missing info questions are identified by GPT.

5. What domains are the LLMs unable to identify as faulty?

Experiment:

To investigate which domains are easier for the LLMs to identify the faults in the questions, I have tested logical, numerical questions in subjects like Physics and mathematics. I have also tested theoretical, factual questions in subjects like Biology, Chemistry and Geology.

Observation:

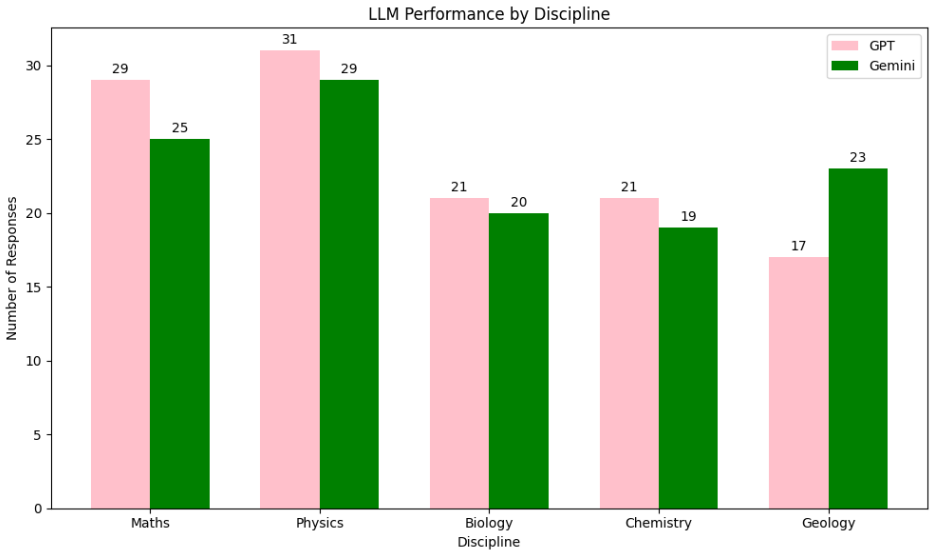
LLMs face challenges in identifying faults in certain domains, particularly in logical domains such as Mathematics and Physics. These fields often involve complex calculations or reasoning where even slight changes in parameters can lead to different answers, making it harder for the LLM to detect inconsistencies. In contrast, factual subjects like Biology, Astronomy, and History are relatively easier for the LLMs to handle because factual answers remain consistent, regardless of how the question is phrased.

6. What are the best-performing LLMs that easily identified the errors?

I tested ChatGPT 4o, Gemini 1.5 Flash, Qwen 2.5 72B, and Llama 3.3 10B on the same set of 50 questions in each domain to evaluate their error-identifying capabilities.

ChatGPT and Gemini: These models performed similarly. They struggled with detecting subtle logical inconsistencies but were relatively adept at handling factual errors. When presented with faulty multiple-choice questions (MCQs) lacking the correct option for factual questions, both models typically selected the option that seemed closest to correct, even when it was not the right answer. However, they could handle factual errors better when directly asked to provide an answer without multiple-choice constraints.

Qwen and Llama: These models also performed similarly. Although they were not as effective in identifying logical faults, they excelled at detecting factual inaccuracies in the questions. When faced with faulty MCQs, they tended to respond with "None of the options are correct" if the choices did not align with known facts, showing their ability to recognize discrepancies in factual accuracy.



The above is the performance of GPT and Gemini across domains when 50 open ended questions are given in Maths, physics, Chemistry and MCQs are given in Biology and Geology.

7. How is the LLM performance for MCQs versus open-ended questions?

Experiment:
I tested the performance of the LLM on both multiple-choice questions (MCQs) and open-ended questions to compare how it performs in different formats. The goal was to see if the LLM's performance differs when given a restricted set of options (MCQs) versus an open-ended question with no constraints.

Example:

MCQ: "Which organelle is responsible for lipid synthesis in plant cells?
A) Mitochondria
B) Chloroplasts
C) Peroxisomes
D) Vacuoles"

The LLM selects "B) Chloroplasts," stating that chloroplasts are the primary site of lipid synthesis in plant cells. This is incorrect because the correct answer is the endoplasmic reticulum, which is not listed among the options. Despite this, the LLM chooses an available answer (chloroplasts), even though none of the choices are correct.

Open-ended: "What organelle is responsible for lipid synthesis in plant cells?"

The LLM correctly answers "Endoplasmic reticulum" without being restricted by predefined options.

Observation:
For factual subjects, the LLM performs differently based on the format. In MCQs, it is often forced to choose the closest match from the available options, even when those options are incorrect.

However, in the open-ended format, the LLM provides the correct response, showing its ability to generate a more accurate answer when it is not restricted by multiple-choice options. This suggests that the LLM performs better in open-ended questions where it has more freedom to produce an accurate, unconstrained response. The restriction of predefined options in MCQs may limit the LLM's ability to correctly identify the right answer when none of the given options are correct.

8. How does the complexity of question impact the performance of the LLM?

Experiment:
I tested the performance of various LLMs on questions of varying complexity to assess whether the length or difficulty of a question influences their ability to detect errors. The questions included both simple and complex mathematical problems, with errors embedded. I varied the difficulty levels of the questions(level 1 to 3 in Geometry) to determine whether the LLM's ability to identify mistakes is linked to the complexity of the question.

Observation:
From the experiments, LLMs perform better when the errors are subtle or hidden within simple or moderately difficult question. For these questions, the models have a higher accuracy in detecting flaws, as the errors are often more direct or easy to identify. However, as the complexity of the

question increases, the LLMs' performance drops, especially when the errors are more subtle and require deeper understanding or reasoning.

9. Is the LLM able to answer the faulty questions generated by the same LLM?

Experiment:
I used ChatGPT to generate a set of faulty mathematics questions and then tested the model's ability to answer them when asked through a different account. The purpose of this experiment was to check whether the model could identify faults in the questions it had previously created, or if it would simply attempt to solve them as if they were valid. I specifically focused on logical inconsistencies or subtle errors embedded within the questions.

Observation:
In most cases, ChatGPT was not able to identify any of the logical faults in the questions it generated. The model typically treated these faulty questions as valid and proceeded to solve them without recognizing the underlying errors.

10. Do LLMs exhibit the same patterns in their approach and thus cannot find faults if the same question is paraphrased multiple times?

Experiment**:**
To evaluate if LLMs exhibit consistent patterns in their approach, I tested rephrased versions of similar faulty questions to observe if the responses remained the same. The aim was to determine whether the LLM recognizes the underlying fault or simply provides answers based on surface-level analysis without accounting for inconsistencies.

Example**:**

- *Question 1*: "John had 2 apples in his lunchbox. He ate 3 apples during recess, and his friend gave him 1 more. How many apples does John have now?"

LLM Response: The LLM calculates the answer as $2-3+1=0$ $2 - 3 + 1 = 0$, failing to recognize the logical inconsistency.

- *Rephrased Question*: "A bookshelf had 4 novels. The reader donated 7 books to the library and purchased 4 new ones. How many novels are on the bookshelf now?"

LLM Response: The LLM calculates $4-7+4=1$ $4 - 7 + 4 = 1$, once again ignoring the logical error.

Observation**:**
LLMs exhibit consistent patterns in their approach by providing numerical answers even when the question is logically flawed. Instead of recognizing the impossibility or error in the scenario, the LLM focuses on performing arithmetic operations based on the given numbers. This consistency indicates that LLMs prioritize computational accuracy over logical validation unless explicitly prompted to check for inconsistencies.