Can LLMs replace Neil deGrasse Tyson?
Evaluating the Reliability of LLMs as Science Communicators

Prasoon Bajpai[1], **Niladri Chatterjee**[1], **Subhabrata Dutta**[2], **Tanmoy Chakraborty**[1]
[1]Indian Institute of Technology, Delhi, [2]TU Darmstadt
prasoonbajpai786@gmail.com, {niladri, tanchak}@iitd.ac.in, subhabrta.dutta@tu-darmstadt.de

Abstract

Large Language Models (LLMs) and AI assistants driven by these models are experiencing exponential growth in usage among both expert and amateur users. In this work, we focus on evaluating the reliability of current LLMs as science communicators. Unlike existing benchmarks, our approach emphasizes assessing these models on scientific question-answering tasks that require a nuanced understanding and awareness of answerability. We introduce a novel dataset, `SCiPS-QA`, comprising 742 Yes/No queries embedded in complex scientific concepts, along with a benchmarking suite that evaluates LLMs for correctness and consistency across various criteria. We benchmark three proprietary LLMs from the OpenAI GPT family and 13 open-access LLMs from the Meta Llama-2, Llama-3, and Mistral families. While most open-access models significantly underperform compared to GPT-4 Turbo, our experiments identify Llama-3-70B as a strong competitor, often surpassing GPT-4 Turbo in various evaluation aspects. We also find that even the GPT models exhibit a general incompetence in reliably verifying LLM responses. Moreover, we observe an alarming trend where human evaluators are deceived by incorrect responses from GPT-4 Turbo.

1 Introduction

The surge of Large language models (LLMs) (Brown et al., 2020; Chowdhery et al., 2022; Chung et al., 2022; OpenAI, 2022) marks the beginning of an era of rapid development across a variety of natural language tasks. With the introduction of chatbots powered by instruction-tuned LLMs, users across diverse domains are becoming reliant on them in day-to-day activities. The increasing usage of LLM-based AI assistants in academia has triggered intense discussion recently. Multiple reports of inconsistent fragments of text appearing in scientific papers, apparently generated by AI assistants and overlooked due to lack of caution, have surfaced. Recent attempts have been made to outline the usage of AI assistants for literature surveys in research pipelines (Bhayana, 2024; Whitfield and Hofmann, 2023)

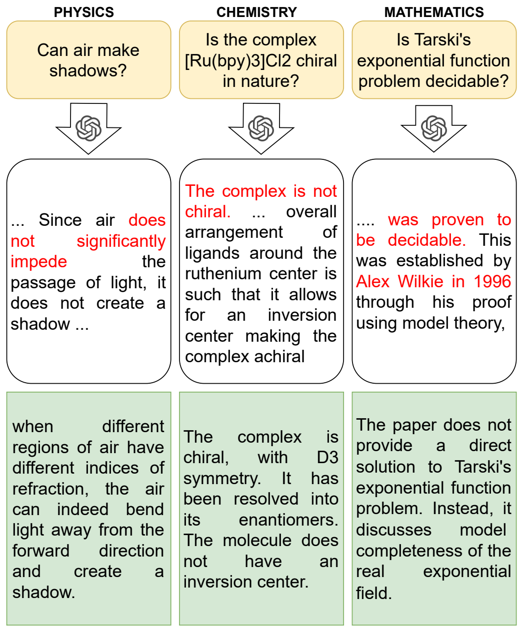


Figure 1: Examples of wrong reasonings given by GPT-4 Turbo to problems in `SCiPS-QA` (white rectangles). The corresponding human-generated correct reasonings are provided in green rectangles: (Physics – Air can cast a shadow under conditions of non-uniform refractive index (Baird, 2024); Chemistry – The complex is chiral with D3 symmetry (Ghosh et al., 1984); Mathematics – The paper discusses the model completeness of the real exponential field and its connection to Tarski's problem and the first root conjecture. Tarski's problem is an open problem (Macintyre and Wilkie, 1996)).

However, there are innate risks associated with LLMs, attributed to overconfident generation and hallucination, that need to be addressed before their large-scale usage as surrogates of human expertise. Particularly in scientific communication in which

Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pages 15895–15912
November 12-16, 2024

nuance plays a vital role, LLMs missing out on small details can spread misconceptions (Dutta and Chakraborty, 2023). Another key challenge lies in the lack of self-awareness of current LLMs and overconfident generation leading to hallucination; given that realizing the lack of knowledge drives the pursuit of scientific exploration, an uncompromising quality of an AI assistant would be to reflect on the lack of knowledge. Existing STEM benchmarks, despite a variety in problem hardness, fail to incorporate these crucial characteristics.

In this work, we seek to close the gaps in evaluating LLMs towards faithful scientific question-answering. Specifically, we seek to address the following research questions:

- **RQ1:** Can existing LLMs answer scientific reasoning questions successfully and faithfully that require understanding the nuances of scientific knowledge?
- **RQ2.** Are LLMs effective at abstaining from assertively answering scientific open problems?
- **RQ3.** Can LLMs successfully verify LLM-generated responses?
- **RQ4.** Can human evaluators be misled by incorrect yet confident LLM responses to complex scientific questions?

To this end, we propose a novel dataset scientific QA dataset, `SCiPS-QA` (**S**pecially **C**hallenging **P**roblems **i**n **S**cience – **Q**uestion **A**nswering), a collection of 742 complex boolean scientific problems that require deep knowledge retrieval and extensive reasoning to answer (**Contribution #1**).[1] The problems are chosen from the most niche research areas across different subjects (see Figure 1 for sample questions from `SCiPS-QA` and answers generated by GPT-4 Turbo). `SCiPS-QA` contains *closed* (i.e., the answer exists within the scope of current scientific knowledge) as well as *open* problems. We benchmark a wide variety of proprietary and open-access LLMs from the OpenAI GPT series, Llama-2 and Llama-3 series, and Mistral series on `SCiPS-QA` using an exhaustive evaluation suit to judge their correctness, faithfulness, and hallucination, in terms of the final boolean answer as well as the reasoning explanation (**Contribution #2**). We find that while proprietary models like GPT-4 Turbo are generally better than open-access Llama-2, Mistral, or smaller Llama-3 variants, Llama-3-70B models (with or without instruction tuning) come as a strong competitor to GPT-4 Turbo (**Findings #1**). However, all the experimented LLMs are far from understanding the nuances of scientific rigor, particularly in relation to open problems (**Findings #2**). We investigate whether proprietary LLMs can successfully verify LLM-generated responses to these complex scientific questions (**Contribution #3**), revealing their shortcomings in verifying different aspects of the generated response (**Findings #3**). Finally, we perform a human evaluation of GPT-4 Turbo generated responses to a subset of questions from `SCiPS-QA` (**Contribution #4**). Alarmingly, the persuasive style of generation adopted by GPT-4 Turbo is enough to deceive human evaluators to trust the reasoning, particularly when answers are included in the response (**Findings #4**).

[1] Please find the code and data at the github repo : llm-science-miscommunication

2 Related Work

LLMs have demonstrated various types of reasoning capabilities, including logical, commonsense, mathematical, and temporal reasoning (Huang and Chang, 2023). In this section, we review relevant work that explores the limitations of LLMs in scientific reasoning.

Several datasets provide comprehensive assessments of LLMs' abilities to solve mathematical problems. GSM8K (Cobbe et al., 2021) comprises high-school-level math word problems, while AQuA-RAT (Ling et al., 2017) includes a collection of algebraic word problems. Dolphin18K (Huang et al., 2016) features elementary-level problems designed to evaluate basic mathematical reasoning capabilities. The MATH dataset (Hendrycks et al., 2021) presents more challenging problems than those in the aforementioned datasets but focuses on simpler mathematical objects compared to the complex scientific concepts found in `SCiPS-QA`. Additionally, Ape210K (Zhao et al., 2020) offers a broad range of mathematical problems to further test LLMs' problem-solving skills. These datasets collectively highlight the strengths and limitations of LLMs in mathematical reasoning, providing a foundation for understanding their performance in more specialized scientific domains.

ScienceQA (Lu et al., 2022), SciQ (Lu et al., 2022) and MMLU (Hendrycks et al., 2020) are prominent datasets used to evaluate LLMs' scientific reasoning capabilities. MMLU-Pro (Wang et al., 2024) is an improved version of MMLU,

Subject	Closed	Open	Total
Physics	195	47	242
Chemistry	132	0	132
Mathematics	140	143	283
Theoretical CS	26	22	48
Astronomy	15	0	15
Biology	1	14	15
Economics	1	6	7
Total	**510**	**232**	**742**

Table 1: Composition of SCiPS-QA.

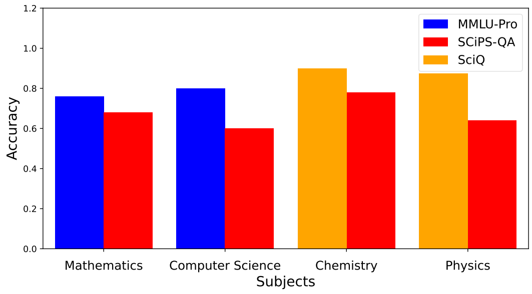


Figure 2: Performance of GPT-4 Turbo on a random subset (of size 40) of MMLU-Pro, SciQ and SCiPS-QA. GPT-4 Turbo performs worst on SCiPS-QA across all subjects.

offering more challenging problems and greater resistance to prompt variations. ScienceQA is a large-scale multimodal dataset with $21,208$ multiple-choice questions covering diverse science topics. In contrast, SciQ comprises 13.7K multiple-choice science exam questions created through crowd-sourcing. We demonstrate that GPT-4 Turbo performs better on these popular STEM datasets than on SCiPS-QA.

SCiPS-QA also focuses on benchmarking answer abstinence in LLMs by including open scientific queries in Physics, Chemistry, and Mathematics. Feng et al., 2024 explored various answer abstinence methods, evaluating them on MMLU. Wen et al., 2024 investigated the ability of LLMs to abstain from answering context-dependent science questions when provided with insufficient or incorrect context. They used datasets such as ScienceQA, OpenBookQA, ARC (AI2 Reasoning Challenge) (Clark et al., 2018), and QASPER (Dasigi et al., 2021) to study LLM abstention behavior. ScienceQA includes school-level and college-level scientific problems requiring relatively simple reasoning capabilities. OpenBookQA features straightforward open-book style general and scientific reasoning problems, and ARC contains grade-school level multiple-choice science questions. In contrast, SCiPS-QA provides a much tougher benchmark for scientific reasoning and explores answer abstinence in Boolean questions.

3 The SCiPS-QA Dataset

In this section, we describe the composition of SCiPS-QA and the methodology used to collect the Boolean queries that constitute SCiPS-QA.

The dataset comprises 742 complex Yes/No problems that require expert-level proficiency in scientific reasoning to answer correctly. We include both open and closed problems across subjects – Physics, Chemistry, Mathematics, Theoretical Computer Science, Astronomy, Economics, and Biology. Table 1 provides the composition of SCiPS-QA, while Figure 5 shows the subject-level topic decomposition. The difficulty of the problems in the dataset is deliberately kept very high to rigorously test the scientific reasoning and Boolean answering capabilities of state-of-the-art open-source and proprietary LLMs.

We randomly select 40 problems from each of four different subjects within SCiPS-QA to compare GPT-4 Turbo's performance in answering Boolean scientific queries against those from MMLU-Pro and SciQ. Additionally, we utilize GPT-3.5 Turbo to paraphrase 40 randomly chosen scientific problems per subject from MMLU-Pro and SciQ into a Yes/No format. Figure 2 illustrates that GPT-4 Turbo performs the worst on SCiPS-QA, highlighting its higher level of difficulty regarding boolean question answering.

Closed questions. These questions have definitive answers supported by scientific literature. We curate a list of complex topics for each subject in SCiPS-QA manually. For each topic, we utilize the `wikipedia` API to retrieve its summary. Subsequently, we provide this summary to GPT-4 Turbo, prompting it to generate Yes/No problems along with their corresponding answers. The resulting Boolean questions undergo manual assessment based on two criteria: (1) requiring scientific reasoning for accurate answers, and (2) correctness of the generated answers. The precise prompt used to generate closed questions can be found in Appendix D.1.

Open questions. These questions lack a definitive answer in the scientific literature. They are manually selected from `wikipedia` pages and research blogs. Further details on how open questions were collected can be found in Appendix B.

4 Experiments

This section presents the details of the experiments we performed to answer the research questions (RQs) we set to explore.

4.1 Experimental Setup

We evaluate a total of 13 open-source models, including those from the Llama-2 family (Touvron et al., 2023), Llama-3 family, Mistral-7B-Instruct-v0.1 (Jiang et al., 2023), Mistral-7B-Instruct-v0.2, and Mistral-8x7B-Instruct-v0.1 (Jiang et al., 2024), on the `SCiPS-QA` dataset using custom-designed evaluation metrics. Additionally, we assess proprietary models such as GPT-4 Turbo (gpt-4-turbo-2024-04-09), GPT-3.5 Turbo (gpt-3.5-turbo-1106), and 'text-davinci-003'. For proprietary models, we follow the methodology outlined in (Li et al., 2024) to evaluate the reasoning passages generated by these models in response to boolean queries from `SCiPS-QA`. Evaluation criteria include attributes like *factuality* and *convincingness* (defined in Appendix A), assessed using GPT-3.5 Turbo and human experts as evaluators. We also evaluate the propensity for *hallucination* (see Appendix A) in these reasoning passages using SelfCheckGPT (Manakul et al., 2023), which employs a sampling-based approach. Further details on these evaluations are provided in subsequent sections. We collect responses through few-shot prompting. The details about exact prompts can be found in Appendix D.2. For each model, the responses are collected in two different settings. We call responses collected at temperature 0.0 the 'main responses' and those collected at temperature 1.0 the 'stochastic responses'.

4.2 Evaluation Metrics

Towards a comprehensive evaluation of LLMs, we define the following metrics on the generated responses.

(i) **Main Response Accuracy (MACC).** The accuracy of responses obtained at zero temperature.

(ii) **Major Stochastic Response Accuracy (MSACC).** We collect the majority response from 10 different stochastic responses with temperature set to 1. We treat invalid responses as incorrect answers.

(iii) **Variation in Stochastic Responses (VSR).** We report the variety in the 10 stochastic responses obtained at temperature 1. We map $A \to 1$, $B \to 2$, $C \to 3$ and rest of the invalid responses to 3 and calculate the standard deviation.

(iv) **Accuracy of Main responses for closed questions (CMACC)** denotes the MACC score on the subset of `SCiPS-QA` containing closed questions.

(v) **Accuracy of Major Stochastic Responses for Closed Questions (CMSACC)** reflects whether the majority of the LLMs' responses to the closed questions in a unit temperature decoding are correct or not.

(vi) **Accuracy of Main Responses for Open Questions (OMACC)** is similar to CMACC but evaluated on the open questions instead. In addition to the overall correctness, this metric evaluates whether the model can identify if a question is scientifically unanswerable.

(vii)**Accuracy of Major Stochastic Responses for Open Questions (OMSACC)** tests the answer abstinence of the LLM in a unit temperature generation regime.

4.3 Hallucination Quantification

We employ SelfCheckGPT (Manakul et al., 2023), a sampling-based methodology that assigns hallucination scores within the range of [0, 1] (0: no hallucination and 1: full hallucination). This scoring is derived by measuring the deviations between the main response and multiple stochastic responses. We take the average of the hallucination scores of sentences in the main response to assign a hallucination score to the entire main response. We briefly describe each of the SelfCheckGPT variants in this section. More details about SelfCheckGPT and the variants we have implemented can be found in the Appendix C.

SelfCheckGPT with BERTScore. For each reasoning sentence in the main response, we calculate the maximum semantic similarity across all sentences in the stochastic response passages. This score indicates the degree of semantic similarity between the main response sentence and various stochastic responses. To quantify hallucination, we derive the complement of this score and assign it as the hallucination score for the main response sentence. Our analysis employs two models, all-MiniLM-L6-v2 and all-mpnet-base-v2, sourced from `sentence_transformer` (Reimers and Gurevych, 2019), to generate sentence-level embeddings. This approach ensures mitigation of potential model bias in our results.

SelfCheckGPT with NLI. Natural Language Inference (NLI) assesses whether a hypothesis logically follows from a premise, categorized as entailment, neutral, or contradiction. We compare each sentence of a main response reasoning passage as a hypothesis against each of the corresponding stochastic response reasoning passages as the premise. The logits associated with classes 'contradiction' and 'entailment' are considered and a score is assigned to the main response sentence, which is a proxy for the probability score of it being in 'contradiction' to the stochastic response reasoning passages. We use DeBERTa-v3-base (He et al., 2020) fine-tuned on MNLI (Williams et al., 2018) for collecting the logits associated with 'contradiction' and 'entailement' classes.

SelfCheckGPT with Prompt. We employ an external LLM evaluator to determine if each sentence in a main response reasoning passage is supported by corresponding stochastic response reasoning passages. Specifically, we utilize GPT-3.5 Turbo as the external LLM; the exact prompt used can be found in Appendix D.3. The responses (Yes, No, NA) are mapped to hallucination scores (Yes $\rightarrow$ 0, No $\rightarrow$ 1, NA $\rightarrow$ 0.5). The average of the GPT-3.5 Turbo response scores is calculated and assigned as the hallucination score for the corresponding sentence in the main response.

4.4 NLG Evaluation of Reasoning Passages

We validate the main response reasoning passages generated by the models – GPT-4 Turbo, GPT-3.5 Turbo, and text-davinci-003 using GPT-3.5 Turbo as the verification model. Additionally, we verify responses from GPT-4 Turbo using GPT-4 Turbo itself as the verification model. Verification attributes are scored on a linear scale using prompt outputs in a zero-shot setting. All relevant prompt details can be found in Appendix D.4.

Convince-factor. Responses that are highly convincing but rely on incorrect information are considered 'hallucinations' (Ji et al., 2022). We assign a *convincingness score* on a linear scale ranging from 1 to 5. This verification attribute is reported for main response reasoning passages using two different prompt settings: one where model answers are included in the prompt given to the evaluator models (denoted as `convince-factor-with-answer`), and another where model answers are absent (denoted as `convince-factor-without-answer`).

Fact-check. We assign scores on a linear scale (ranging from 1 to 5) to main response reasoning passages based on their factual accuracy. Our aim is to investigate whether evaluator LLMs can differentiate between incorrect reasoning passages and correct ones based on the factual correctness of responses.

Information Mismatch. We compare each main response reasoning passage with all ten different stochastic response reasoning passages S_k for the amount of information mismatch between them, which is scored on a linear scale ranging from 1 to 5. We assign the mean of such scores across stochastic responses to the main response reasoning passage.

4.5 Human Evaluations

We randomly select 30 combinations of query and main response reasoning passages (from GPT-4 Turbo) for each subject – Physics, Chemistry, Mathematics, and Computer Science. For each subject, we employed two human evaluators. All human evaluators had at least a graduate degree in their respective subjects; they were male and aged between 20-25.

Human evaluators were tasked with assigning a 'convince-factor' score to the main response reasoning passages, following the same evaluation setup used with LLMs as the evaluators. We divide human evaluators into two groups: one group sees both the model answer and reasoning, while the other group only views the reasoning itself. Both groups receive identical queries for evaluation.

5 Results

In this section, we look at the various quantitative results summarized in Table 2.

5.1 `SCiPS-QA` Benchmark

We observe that among both open-source and proprietary models, the Llama-2 family consistently performs the poorest across nearly all metrics. The GPT series of models show competitive performance, closely rivaling the higher-scale models within the Llama-3 family, which rank highest among the open-source models tested.

MACC: Llama-3-70B achieves the highest score in the MACC metric at 0.693, closely followed by GPT-4 Turbo with a score of 0.646. No-

LLMs	MACC (↑)	MSACC (↑)	VSR (↓)	CMACC (↑)	CMSACC (↑)	OMACC (↑)	OMSACC (↑)
meta-llama-2-7B	0.021	0.108	0.922	0.031	0.157	0.000	0.000
meta-llama-2-7B-chat	0.321	0.272	1.069	0.284	0.255	0.400	0.310
meta-llama-2-13B	0.327	0.361	0.826	0.476	0.523	0.000	0.004
meta-llama-2-13B-chat	0.341	0.356	0.636	0.484	0.500	0.026	0.039
meta-llama-2-70B	0.532	0.274	1.097	0.498	0.292	0.608	0.232
meta-llama-2-70B-chat	0.423	0.426	0.689	0.616	0.620	0.000	0.000
meta-llama-3-8B	0.120	0.010	1.014	0.174	0.139	0.000	0.004
meta-llama-3-8B-instruct	0.444	0.437	0.550	0.645	0.635	0.004	0.000
meta-llama-3-70B	**0.693**	0.605	0.964	0.743	0.659	**0.582**	**0.487**
meta-llama-3-70B-instruct	0.628	<u>0.623</u>	0.295	**0.780**	**0.784**	0.293	0.267
Mistral-7B-Instruct-v0.1	0.113	0.311	0.660	0.165	0.453	0.000	0.000
Mistral-7B-Instruct-v0.2	0.496	0.488	0.474	0.582	0.574	0.306	0.297
Mixtral-8x7B-Instruct-v0.1	0.591	0.596	0.555	0.678	0.682	0.401	0.405
text-davinci-003	0.548	0.554	<u>0.229</u>	0.723	0.717	0.187	0.216
GPT-3.5 Turbo	0.576	0.597	0.337	0.691	0.711	0.340	0.361
GPT-4 Turbo	<u>0.646</u>	**0.651**	**0.193**	<u>0.750</u>	<u>0.754</u>	<u>0.432</u>	<u>0.436</u>

Table 2: Comparative evaluation of state-of-the-art open-source and proprietary LLMs across multiple evaluation metrics. The symbol ↑ (↓) indicates the higher (lower) value is better. We **bold** the best and <u>underline</u> second-ranked score for each metric.

tably, among the Llama-2 and Llama-3 families, 'chat' models perform equivalently to their non-instruction fine-tuned counterparts, except for the lower scale members: Llama-2-7B and Llama-3-8B, where the instruction fine-tuned variants show score increases of 0.3 and 0.324, respectively. Mixtral-8x7B-Instruct-v0.1 significantly outperforms both Mistral-7B-Instruct-v0.1 and Mistral-7B-Instruct-v0.2. All three GPT models perform strongly, with GPT-4 Turbo achieving the highest score of 0.646 in the MACC metric.

MSACC: GPT-4 Turbo outperforms all other models with a score of 0.651, closely followed by Llama-3-70B-instruct, which achieves a score of 0.623. In contrast to the MACC metric, where instruction fine-tuned models from the Llama-2 and Llama-3 families often performed equivalent to their non-instruction fine-tuned counterparts, here we observe that the instruction fine-tuned models outperform their counterparts.

VSR: The Llama-2 family performs the worst in terms of VSR score indicating their limited capability to produce consistent results. In contrast, the GPT models exhibit high consistency, with GPT-4 Turbo reporting the lowest VSR score of 0.193 among all models. The Llama-3 family demonstrates better consistency compared to the Llama-2 family, while Mistral models also perform well but not as strongly as the top performers among open-source models. Among them, Llama-3-70B-instruct stands out with a VSR score of 0.295.

CMACC, CMSACC: Llama-3-70B-instruct outperforms all models in CMACC and CMSACC metrics achieving a score of 0.780 and 0.784 respectively. GPT models also perform well in handling closed domain scientific queries with GPT-4 Turbo being the best among them, achieving a score of 0.75 and 0.754.

OMACC, OMSACC: One of the major findings is that most of the open source and proprietary LLMs are really bad at accepting that they do not know the answers to open scientific queries in `SCiPS-QA`. This is evident from their low OMACC and OMSACC scores across the board. Llama-3-70B stands out as the top performer in terms of answer abstention for open scientific queries, achieving the highest OMACC (0.582) and OMSACC (0.487) scores. In contrast, Llama-2 models struggle significantly in handling open queries, while Mistral-7B models and Mixtral-8x-7B-Instruct-v0.1 perform reasonably well among open models. The GPT models demonstrate strong performance in responding to open scientific queries, with GPT-4 Turbo achieving the highest scores of 0.432 and 0.436 in OMACC and OMSACC metrics, respectively. Note that models also produced invalid responses to prompts. Small models – Llama-2-7B and Mistral-7B-Instruct-v0.1, produce a much larger fraction of invalid responses as compared to other open-source models. Proprietary models produce almost negligible invalid responses, with GPT-4 Turbo reporting no invalid main response. More details can be found in Appendix E.1.

5.2 Hallucination Quantification

Our investigation using SelfCheckGPT fails to yield conclusive evidence of hallucination in the proprietary GPT models despite their high rate of

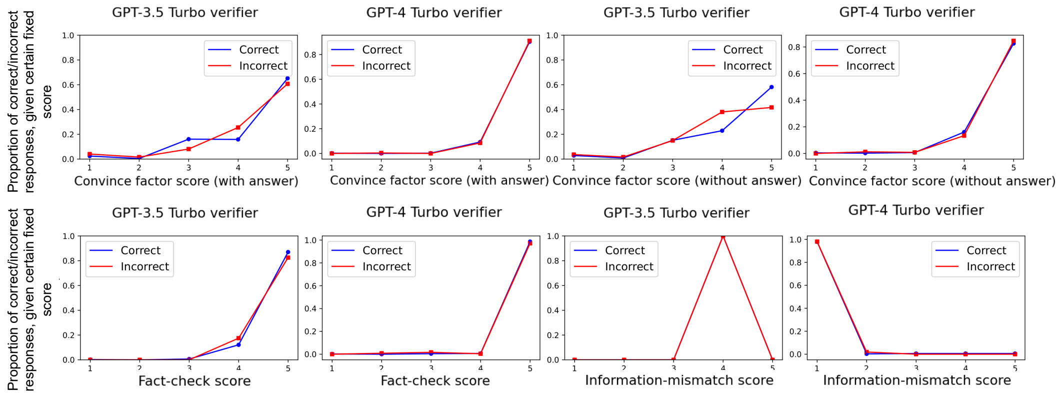


Figure 3: Verification of the reasoning passages generated by GPT-4 Turbo across convincingness (with and without answer), factuality, and information mismatch; we use both GPT-4 Turbo and GPT-3.5 Turbo as verifier models. The fraction of correct (incorrect) responses at each score level is shown in blue (red). An ideal verifier should provide all the incorrect responses with the lowest score (1) and all the correct responses with the highest score (5). However, no verifier model in our experiments could demarcate between the correct and incorrect responses.

mistakes. When employing the BERTScore variant, we observe normal distribution in the frequency distribution histograms (Figure 6) for all three GPT models on `SCiPS-QA`. Interestingly, GPT-3.5 Turbo achieves the lowest mean hallucination score, followed by GPT-4 Turbo, while text-davinci-003 performed the poorest.

Assuming normal distribution and independence of score samples, we conduct Welch's t-tests to assess the statistical significance of mean differences between the proprietary models. Our findings indicate that we fail to reject the hypothesis of no difference in means between all pairs of proprietary models being tested with a 95% level of confidence. Further details of these tests are available in Appendix E.2.

Figure 8 summarizes results for SelfCheckGPT with Prompt. GPT-3.5 Turbo shows a higher count of low hallucination scores given to queries in `SCiPS-QA` than GPT-4 Turbo.

5.3 NLG Evaluation of Reasoning Passages

We assess the main response reasoning passages from all three proprietary models using GPT-3.5 Turbo as the verifier. Table 7 shows the convincingness (with and without the answer), factuality and information-mismatch scores for all three models using GPT-3.5 Turbo as a verifier. We use GPT-4 Turbo and GPT-3.5 Turbo to verify reasoning passages obtained from GPT-4 Turbo. Figure 3 shows the verification results for the GPT-4 Turbo's reasoning passages for the two verifier models.

5.3.1 Convince-factor

GPT-3.5 Turbo consistently assigns high scores to the main response reasoning passages from all three models (see Table 7). It rates both correct and incorrect reasoning responses highly across all models. Surprisingly, even when evaluating reasoning passages from GPT-4 Turbo, it itself struggles to distinguish between correct and incorrect responses. Interestingly, as depicted in Figure 3, GPT-4 Turbo assigns a higher fraction of reasoning passages (both correct and incorrect) a perfect score of 5 in convincingness (with and without answer) compared to GPT-3.5 Turbo. This suggests that GPT-4 Turbo performs worse than GPT-3.5 Turbo in terms of verifying its responses based on convincingness (with and without answer).

5.3.2 Fact-check

GPT-3.5 Turbo assigns high scores to the reasoning passages from all three models (see Table 7 in Appendix), often rating a majority of incorrect responses a perfect 5 in factuality verification. GPT-4 Turbo performs even worse in verifying its own reasoning passages (see Figure 3), assigning a higher fraction of incorrect reasoning passages a perfect 5 score compared to GPT-3.5 Turbo. This indicates that GPT-4 Turbo struggles more than GPT-3.5 Turbo in distinguishing between correct and incorrect reasoning passages, even when evaluating its own responses.

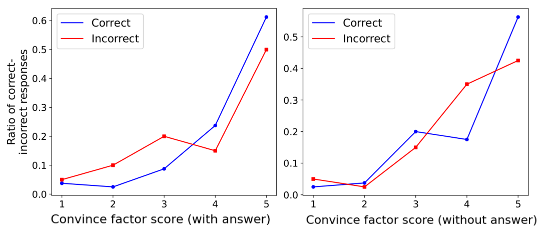


Figure 4: Distribution of correct (in blue) and incorrect (in red) responses generated by GPT-4 Turbo against convince factor scores provided by human evaluators. Incorrect LLM reasoning can deceive humans as convincing with or without the answer shown to them. However, humans provide better judgement with the answer.

5.3.3 Information Mismatch

We observe that GPT-3.5 Turbo assigns relatively high `information-mismatch` scores to main response reasoning passages from all three proprietary models. Table 7 shows that among the three models being tested, GPT-3.5 Turbo gives a lesser information-mismatch score to its own reasoning passages than it does to the other two models.

From Figure 3, we observe that GPT-4 Turbo provides a very low score for its own reasoning passages in terms of `information-mismatch` score. These patterns are agnostic to the correctness of reasoning passages, suggesting that both verifier models are not able to differentiate between correct and incorrect passages using `information-mismatch` scores. Consistent with our observations across various verification attributes, GPT-4 Turbo performs worse than GPT-3.5 Turbo, consistently assigning lower scores (often 1) to most reasoning passages, irrespective of their correctness.

5.4 Human Evaluations

Human evaluators typically fare better than LLM evaluators. As we can see in Figure 4, correct responses are consistently given better scores than incorrect ones. However, a considerable fraction of incorrect responses can still deceive human judgment into getting scores greater than 3.

Notably, human evaluators tend to judge incorrect responses better when the generated answer is attached. This can be possibly related to cases where the LLM infers incorrect answers even after providing correct reasoning context. Furthermore, correct responses are typically distributed towards the highest convince factor (i.e., 5); although, without the answer provided, some correct responses are given scores as low as 3. Interestingly, the scoring distribution provided by human evaluators is much closer to that provided by GPT-3.5 Turbo as verifier instead of GPT-4 Turbo.

6 Qualitative Analysis

In this section, we unveil some of the common error patterns we observed in the reasoning passages generated from GPT-4 Turbo. While answering open problems, we often found GPT-4 Turbo misinterpreting some key scientific relations in questions, hence generating wrong reasoning passages. For the following open problem - *'Question - Is every finitely presented periodic group finite?'*, GPT-4 Turbo responded 'Notably, there are examples of infinite, finitely presented periodic groups, such as the Tarski monster, which is an infinite group where every proper subgroup is cyclic of a fixed prime order. This example shows that it is possible for a finitely presented periodic group to be infinite'. Tarski monster is not a finitely presented periodic group but it is a finitely generated periodic group. GPT-4 Turbo misinterprets relations when dealing with complex scientific subjects in the query and produces reasoning passages for a wrong hypothesis.

Another observation was a lack in reporting contrasting theories to explain a scientific idea. GPT-4 Turbo used only one set of theoretical assumptions to answer the query. For question - *'Is the universe homogeneous and isotropic at large enough scales?'*, GPT-4 Turbo responded - 'According to the cosmological principle, the universe is considered to be homogeneous and isotropic at sufficiently large scales'. Findings (commonly called Axis of Evil) have produced newer studies(Copi et al., 2006) refuting the cosmological principle proclaiming that the question does not admit an answer. GPT-4 Turbo does not take into account such research instances.

7 Discussion and Conclusion

Our experiments on `SCiPS-QA` with a diverse array of LLMs using a comprehensive evaluation strategy reveal several key insights. Firstly, existing LLMs, whether open-access or proprietary, demonstrate a limited understanding of scientific methodologies required to serve as reliable assistants. While the parameter scaling law holds within each LLM family, models of similar size across different families are not directly comparable. For instance, Meta Llama-3 70B models emerge as formidable competitors to much larger GPT models, frequently outperforming GPT-4 Turbo in our evaluations. This reiterates earlier findings that parameter scaling alone does not reflect the capabilities of LLMs and current models, along with their training methodology, are underperforming their 'true' potential (Hoffmann et al., 2022).

Echoing Huang et al. (2024)'s findings, we observe that powerful LLMs such as GPT-4 Turbo and GPT-3.5 Turbo struggle to reliably verify their responses. Hallucination detection techniques like SelfCheckGPT also prove ineffective in detecting incorrect reasoning posed by strong LLMs like GPT-4 Turbo in complex questions within SCiPS-QA. In fact, we notice a counterintuitive trend where GPT-3.5 Turbo assigns lower scores to incorrect responses compared to the stronger GPT-4 Turbo.

However, the most concerning finding of this paper revolves around how human evaluators perceive LLM-generated scientific reasoning. When tasked with evaluating the convincingness of reasoning explanations generated by GPT-4 Turbo, human evaluators tend to assign higher ratings to a significant majority of incorrect answers. This aligns with the concern raised by Dutta and Chakraborty (2023) that current LLM-based AI assistants have the potential to propagate widespread scientific misunderstandings if left unchecked.

Implications for future research. We hope that our proposed dataset, SCiPS-QA, along with the evaluation suit we design in this work, will serve as a valuable benchmark for future LLM research. Given the growing popularity of generalist as well as domain-specific AI assistants, we envision a positive future focus in building reliable scientific assistants. Finally, our findings with human evaluation calls upon further focus in trustworthy AI research.

8 Limitations

Boolean format of scientific questions has been adopted in SCiPS-QA. Having a long-text reasoning evaluation while maintaining the complexity of scientific objects should provide a stronger test for evaluating scientific communication. For this, SCiPS-QA needs to be augmented with golden reasoning passages provided by human experts. There is also a need to add more diverse topics to the SCiPS-QA, particularly in Physics, which is dominated by Quantum Mechanics (Appendix 5). There is also an issue of some queries in SCiPS-QA lying outside the knowledge cutoff of some models, making it difficult to accurately assess their reasoning capabilities. Human evaluations may be slightly limited because they do not include highly experienced evaluators in the respective subjects. The testing of reasoning passages from open-source models has also not been done as part of our analysis.

9 Ethical Considerations

The participants in human evaluation were not coerced into participating and were given clear and comprehensive information about the research before they provided informed consent. The identities of the human evaluators have been protected by ensuring their responses cannot be linked back to the specific individuals. The research results are communicated honestly and credibly, and transparency has been maintained throughout the research process.

Acknowledgments

The authors would like to acknowledge the financial support of DYSL-AI.

References

Christopher Baird. 2024. Can air make shadows?

Rajesh Bhayana. 2024. Chatbots and large language models in radiology: A practical primer for clinical and research applications. *Radiology*, 310(1):e232756.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, and Dhariwal et al. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, and Paul Barham et al. 2022. Palm: Scaling language modeling with pathways.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, and Mostafa Dehghani et al. 2022. Scaling instruction-finetuned language models.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the AI2 reasoning challenge. *CoRR*, abs/1803.05457.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *CoRR*, abs/2110.14168.

Craig Copi, Dragan Huterer, Dominik Schwarz, and Glenn Starkman. 2006. On the large-angle anomalies of the microwave sky. *Monthly Notices of the Royal Astronomical Society*, 367:79 – 102.

Pradeep Dasigi, Kyle Lo, Iz Beltagy, Arman Cohan, Noah A. Smith, and Matt Gardner. 2021. A dataset of information-seeking questions and answers anchored in research papers. *CoRR*, abs/2105.03011.

Yue Dong, Shuohang Wang, Zhe Gan, Yu Cheng, Jackie Chi Kit Cheung, and Jingjing Liu. 2020. Multi-fact correction in abstractive text summarization. *arXiv preprint arXiv:2010.02443*.

Subhabrata Dutta and Tanmoy Chakraborty. 2023. Thus spake chatgpt. *Commun. ACM*, 66(12):16–19.

Shangbin Feng, Weijia Shi, Yike Wang, Wenxuan Ding, Vidhisha Balachandran, and Yulia Tsvetkov. 2024. Don't hallucinate, abstain: Identifying llm knowledge gaps via multi-llm collaboration.

Katja Filippova. 2020. Controlled hallucinations: Learning to generate faithfully from noisy data. *arXiv preprint arXiv:2010.05873*.

Pushpito K Ghosh, Bruce S Brunschwig, Mei Chou, Carol Creutz, and Norman Sutin. 1984. Thermal and light-induced reduction of the ruthenium complex cation ru (bpy) 33+ in aqueous solution. *Journal of the American Chemical Society*, 106(17):4772–4783.

Ivan Habernal and Iryna Gurevych. 2016. What makes a convincing argument? empirical analysis and detecting attributes of convincingness in web argumentation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1214–1223, Austin, Texas. Association for Computational Linguistics.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced BERT with disentangled attention. *CoRR*, abs/2006.03654.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *CoRR*, abs/2009.03300.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. 2022. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*.

Danqing Huang, Shuming Shi, Chin-Yew Lin, Jian Yin, and Wei-Ying Ma. 2016. How well do computers solve math word problems? large-scale dataset construction and evaluation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 887–896, Berlin, Germany. Association for Computational Linguistics.

Jie Huang and Kevin Chen-Chuan Chang. 2023. Towards reasoning in large language models: A survey.

Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. 2024. Large language models cannot self-correct reasoning yet.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Yejin Bang, Andrea Madotto, and Pascale Fung. 2022. Survey of hallucination in natural language generation. *CoRR*, abs/2202.03629.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, and Guillaume Lample et al. 2023. Mistral 7b.

Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, and Florian Bressand et al. 2024. Mixtral of experts.

Zhen Li, Xiaohan Xu, Tao Shen, Can Xu, Jia-Chen Gu, and Chongyang Tao. 2024. Leveraging large language models for nlg evaluation: A survey.

Wang Ling, Dani Yogatama, Chris Dyer, and Phil Blunsom. 2017. Program induction by rationale generation: Learning to solve and explain algebraic word problems. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 158–167, Vancouver, Canada. Association for Computational Linguistics.

Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. 2022. Learn to explain: Multimodal reasoning via thought chains for science question answering. In *Advances in Neural Information Processing Systems*, volume 35, pages 2507–2521. Curran Associates, Inc.

Angus Macintyre and A. J. Wilkie. 1996. On the decidability of the real exponential field.

Potsawee Manakul, Adian Liusie, and Mark J. F. Gales. 2023. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. 2020. On faithfulness and factuality in abstractive summarization. *arXiv preprint arXiv:2005.00661*.

OpenAI. 2022. Chatgpt: Optimizing language models for dialogue. openai.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. 2024. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark.

Bingbing Wen, Bill Howe, and Lucy Lu Wang. 2024. Characterizing llm abstention behavior in science qa with context perturbations.

Sharon Whitfield and Melissa A Hofmann. 2023. Elicit: Ai literature review research assistant. *Public Services Quarterly*, 19(3):201–207.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Wei Zhao, Mingyue Shang, Yang Liu, Liang Wang, and Jingming Liu. 2020. Ape210k: A large-scale and template-rich dataset of math word problems.

A Definitions

Hallucination: The generated content that is nonsensical or unfaithful to the provided source input (Filippova, 2020; Maynez et al., 2020), where the source input changes as the task. We take the world knowledge as the source input in our case.
Factuality: Factuality refers to the property of quality of being actual or based on fact (Dong et al., 2020). In our work, we take "facts" as the world knowledge.
Convincingness: Convincingness refers to the ability of a model to effectively influence the audience through language (Habernal and Gurevych, 2016).

B Collection of Open Questions

We collect open-questions from List of unsolved problems article on wikipedia for all subjects. We also referred to the page List of open questions in theoretical computer science by Antoine Amarilli. We use GPT-3.5 Turbo to parse some of the entries on these web pages into a question format.

C SelfCheckGPT

C.1 Notation

We obtain two types of responses from proprietary models for quantifying hallucination. Let M, calling it the 'main response', denote the reasoning passage obtained at temperature 0.0. We sample N = 10 different stochastic responses:$\{S_1, S_2, \ldots S_N\}$, each at temperature 1.0 using the same prompt structure, aiming to measure commonalities between the stochastic responses and the main response. We use SelfGPTCheck to assign a hallucination score to ith sentence of the main response $M_i : H(M_i) -> [0.0, 1.0]$, with 0.0 score given to such sentences that are completely faithful to source input and 1.0 if they are fully hallucinated. The following subsections describe the variants of SelfCheckGPT briefly that we have used in this paper.

C.2 SelfCheckGPT with BERTScore

Let M_i and S_j^k denote the i-th sentence of the main response and the j-th sentence of the k-th stochastic response. Note all these responses are reasoning passages that are provided by the proprietary models tested. We assign a hallucination score to M_i depending on the BERTScore between M_i and S_j^k as follows:

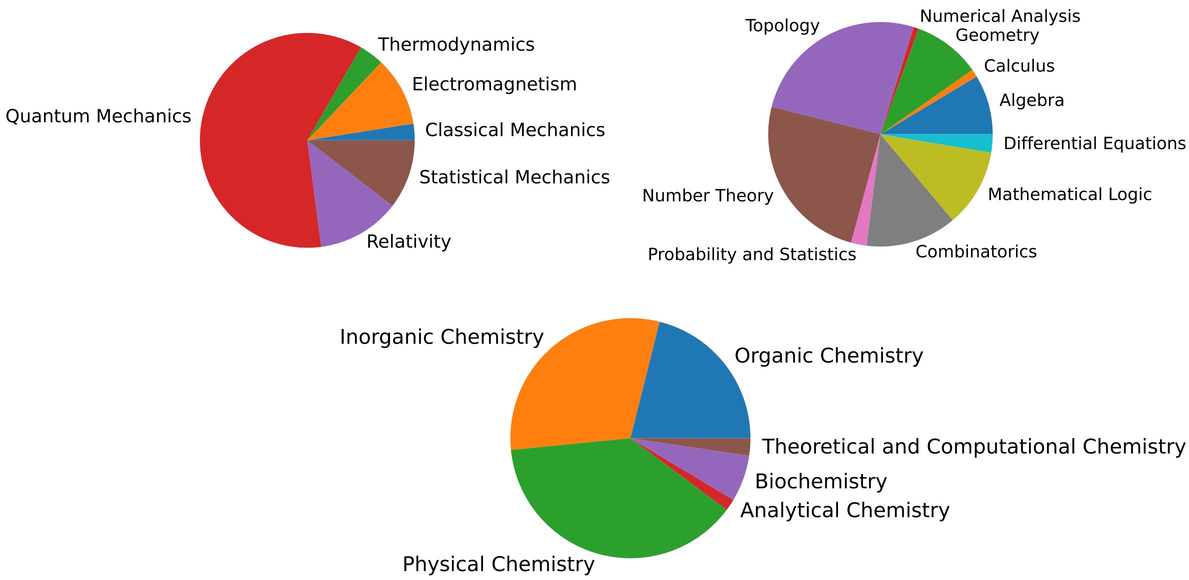


Figure 5: Topic decompostion for subjects : Physics (top-left), Chemistry (middle) & Mathematics (top-right) in `SCiPS-QA`

$$H(M_i) = 1 - \frac{1}{N}\sum_{k=1}^{N} \max_k B(M_i, S_j^k) \quad (1)$$

where $B(.,.)$ is the dot score of sentence embeddings generated using model B. This way M_i shall be assigned a higher score if it is semantically less similar (according to BERTScore) to most of the sentences in different stochastic responses. However, if a sentence in the main response is semantically similar (or appears in) to sentences in different stochastic responses, then it will be assigned a lower hallucination score. We take the mean of the hallucination scores of each sentence of the main response to assign it a hallucination score.

We report results using two different models : $B \in$ {`all-MiniLM-L6-v2`, `all-mpnet-base-v2`} from `sentence_transformer` (Reimers and Gurevych, 2019) for generating sentence-level embeddings for eliminating any possible model bias.

C.3 SelfCheckGPT with NLI

The input for NLI classifiers is typically the premise concatenated to the hypothesis, which for our methodology is the sampled passage S_k concatenated to the sentence to be assessed M_i. Only the logits associated with the 'entailment' and 'contradiction' classes are considered, We use `DeBERTa-v3-base` fine-tuned on MNLI for collecting the logits associated with 'contradict' class.

SelfGPTCheck with NLI uses stochastic response S_k as the premise concatenated to the main response sentence M_i to be assessed. The logits associated with token 'contradict' are used to assign a score.

$$P(\text{contradict}|M_i, S_k) = \frac{\exp(z_c)}{\exp(z_e) + \exp(z_c)} \quad (2)$$

where z_e and z_c are the logits of the 'entailment' and 'contradiction' classes, respectively. A higher probability denotes that the concerned main response sentence disagrees with the stochastic sample and hence, should be assigned a higher hallucination score, which is defined as,

$$H(M_i) = \frac{1}{N}\sum_{k=1}^{N} P(\text{contradict}|M_i, S_k) \quad (3)$$

We take the average of the hallucination scores of sentences in the main response to assign a hallucination score to the entire main response M.

C.4 SelfCheckGPT with Prompt

We prompt GPT-3.5 Turbo to assess if the i-th sentence of the main response is supported by the k-th stochastic response, S_k. The exact prompt can be found in the appendix D.3.

The output from prompting when comparing the i-th sentence against sample S_k is converted to score x_i^k through the mapping Yes: 0.0, No: 1.0, N/A: 0.5. The final inconsistency score is then calculated as:

$$H(M_i) = \frac{1}{N}\sum_{k=1}^{N}(x_i^k) \tag{4}$$

Note, for all these variants, we report the results at only such data-points of `SCiPS-QA` where all 10 stochastic reponses are non-empty and valid. A stochastic response is considered invalid if it cannot be parsed into the boolean answer and the corresponding reasoning passage.

D Prompts

We shall now describe the exact prompts that we used.

D.1 Collection of Closed Questions

We collect closed questions by prompting GPT-4 Turbo to create boolean problems from the passage given in the prompt. The passage is taken from the `wikipedia` pages of topics under different subjects. Table 3 shows the exact prompt that we used for collecting closed questions for `SCiPS-QA`. We replace the `<PASSAGE>` placeholder with the passages retrieved from `wikipedia`.

We observed that most of the questions created by GPT-4 Turbo in this manner, we purely a test of knowledge retrieval. This made us include some additional instructions in the prompt. We manually checked the questions for their corresponding answers and ensured that most of the questions in `SCiPS-QA` required some levels of reasoning to answer.

D.2 Collecting Responses

We now describe the prompts that we used for collecting responses from open-source models and proprietary models.

D.2.1 Open-source Models

Table 3 shows the exact prompts that we used for collecting responses (A- Yes, B - No & C - I do not know) from open-source models. Table 5 shows the number of responses from each open-source model that were invalid. A response (main or stochastic) is considered to be invalid if it could not parsed into one of the choices (A, B or C). We observe that low-scale models Llama-2-7B, Llama-3-8B and Mistral-7B-Instruct-v0.1 had a high percentage of invalid main responses. The instruction fine-tuned versions of models reported much lesser invalid responses at same scale of parameters. The GPT line of models and higher scaled members of Llama-2 and Llama-3 family reported much less percentage of invalid responses (both 'main' and 'stochastic').

While collecting responses from open-source models, we set the generation parameter `max_new_tokens` to 3 and parse the responses for options from the set {A, B, C}, (A - "Yes", B - "No", C - "I do not know"). For models : Llama-2-70B, Llama-2-70B-chat, Llama-3-70B and Llama-3-70B-instruct, we use non-uniform 4-bit quantization to fit these models within a single A100 to account for limited computational resources. Since we also collect reasoning passages from chosen proprietary models, we set the generation parameter `max_tokens` to 1000.

D.2.2 Proprietary Models

The prompt structure for proprietary models differs from that for open-source models with respect to the presence of 'Reason:' field in the exemplars. This is done to force these models to provide reasoning passages which are further quantified for hallucination and score for different attributes using human experts and GPT-3.5 Turbo as evaluators in a parallel setting.

D.3 SelfCheckGPT with Prompt

Table 4 shows the exact prompt that we used for this variant of SelfCheckGPT. The prompt is exactly same mentioned in the SelfCheckGPT paper (Manakul et al., 2023). The `<CONTEXT>` is replaced by each of the stochastic response passages and `<SENTENCE>` is replaced by the main reasoning passages.

D.4 NLG Evaluation of Reasoning Passages

We describe all the prompts that we used for this section. Note that we used GPT-4 Turbo, GPT-3.5 Turb and text-davinci-003 as the LLM modules for assigning scores to the main response reasoning passages.

D.4.1 Convince-factor

Table 4 shows the prompt that we used for two schemes : `convince-factor-with-answer` and `convince-factor-without-answer`. The two prompts differed only with respect to the presence of the model answer (to the boolean scientific query).

D.4.2 Fact-check

Table 4 shows the prompt that we used for assessing the factuality of main response reasoning passages (which replaced the <SOURCE> placeholder)

D.4.3 Information-mismatch

Table 4 shows the prompt that we used for assigning scores of this attribute. The <SOURCE> placeholder is replaced with the main response reasoning passage and the <GENERATED> placeholder is replaced with the stochastic response reasoning passages.

E Results

E.1 Invalid Responses

Table 5 shows the percentage of invalid responses (to the 'answer' field of the prompt) to queries in SCiPS-QA. Llama-3 models and GPT models show fairly low numbers of invalid responses. Low scale models from Llama-2, Llama-3 and Mistral family report high percentage of invalid responses.

E.2 Hallucination Quantification

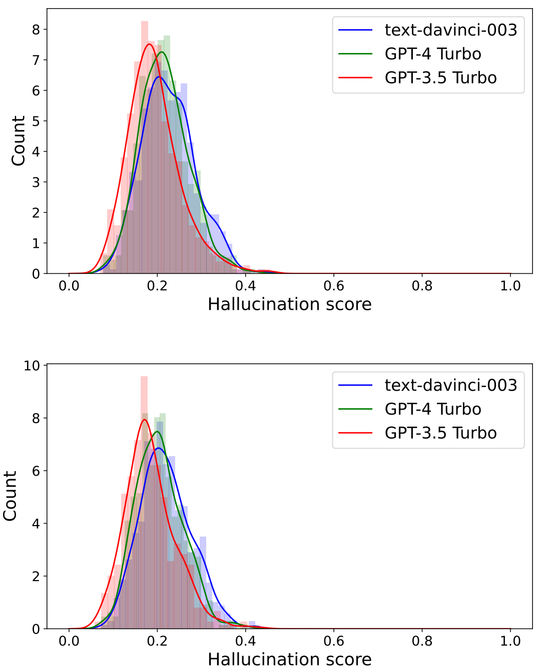


Figure 6: Frequency distribution plots of 'SelfCheckGPT with BERTScore' hallucination scores to main response reasoning passages for sentence_transformer models: all-MiniLM-L6-v2 (above) & all-mpnet-base-v2 (below)

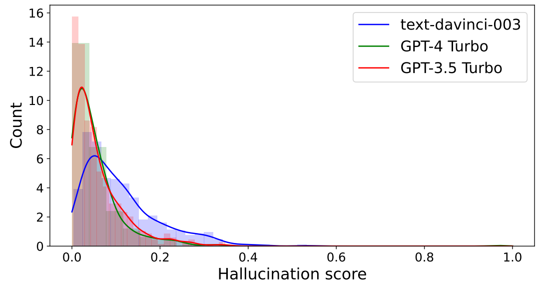


Figure 7: Frequency distribution plots of 'SelfCheckGPT with NLI' hallucination scores to the main response reasoning passages.

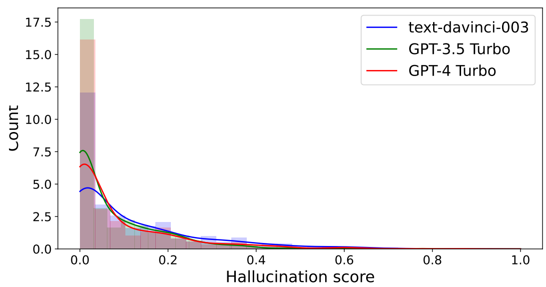


Figure 8: Frequency distribution plots of 'SelfCheckGPT with Prompt' hallucination scores to the main response reasoning passages.

E.2.1 SelfCheckGPT with NLI

Figure 7 shows that main response passages from GPT-3.5 Turbo and GPT-4 Turbo are not demarcated for amount of hallucination using this scoring. text-davinci-003 however, is clearly shown to produce more hallucinated text.

E.2.2 SelfCheckGPT with Prompt

Figure 8 shows results for SelfCheckGPT with Prompt. More response passages from GPT-3.5 Turbo are given low hallucination scores as compared to those from GPT-4 Turbo.

E.2.3 SelfCheckGPT with BERTScore

We performed Welch's t-tests to test the statistical significance of results. We observe that main response reasoning passages from GPT-3.5 Turbo are given least mean hallucination scores using 'SelfCheckGPT with BERTScore' and main response reasoning passages from text-davinci-003 are given the highest mean hallucination scores. We confirm this with Welch's t-tests conducted using `scipy.stats.ttest_ind`:

Notation : Let $\mu_{gpt-4-turbo}$, $\mu_{gpt-3.5-turbo}$ & $\mu_{text-davinci-003}$ represent the sample means of hallucination scores. The details of the tests are present in Table 6

Collecting closed-questions from wikipedia passage†	You are an AI assistant to create extremely challenging Yes/No problems , from the provided passage.
	<PASSAGE>
	Generate your response strictly in the following JSON format.
	Create questions which specifically require reasoning to answer.
	{
	"questions":
	[
	{
	"question" : Can a regular expression represent all possible languages over an alphabet?,
	"answer" : No
	},
	{
	"question" : Can every problem in the complexity class EXP be solved by a deterministic Turing machine in exponential time?,
	"answer" : Yes
	},
	{
	"question" : Is it possible to efficiently approximate the solution to an NP-hard optimization problem within a constant factor?,
	"answer" : No
	}
	]
	}
	
	Create extremely challenging Yes/No questions, requiring reasoning to answer, from the passage provided below. Remember following points -
	1. Use the JSON format specified above.
	2. Create questions that specifically require some reasoning for their resolution.

Collecting responses (open-source models) ‡	You are an AI assistant being used for research purposes. Answer the following Yes/No multi-choice questions with following options. A - Yes B - No C - I do not know Question: Is the nature of spacetime at the Planck scale continuous or discrete? Answer: C Question: Is the exact value of mean lifetime of a neutron known? Answer: B Question: Does the set of natural numbers have the same cardinality as set of integers? Answer: A Question: Is the Universe infinite in spatial extent? Answer: C
Collecting responses (proprietary models)‡	You are an AI assistant being used for research purposes. Answer the following Yes/No multi-choice questions with following options. A - Yes B - No C - I do not know Question: Is the nature of spacetime at the Planck scale continuous or discrete? Reason: The nature of spacetime at the Planck scale is currently not definitively established, and it remains an open question in theoretical physics. However, many theories, including some formulations of quantum gravity, suggest that spacetime may exhibit discrete or quantized characteristics at the Planck scale. This is based on the idea that classical notions of smooth, continuous spacetime may break down at extremely small scales. The uncertainty principle from quantum mechanics also contributes to this speculation. Research in areas such as loop quantum gravity and string theory explores these possibilities, but a conclusive answer is yet to be determined through experimental verification. Answer: C Question: Is the exact value of mean lifetime of a neutron known? Reason: ... Answer: B Question: Does the set of natural numbers have the same cardinality as set of integers? Reason: ... Answer: A Question: Is the Universe infinite in spatial extent? Reason: ... Answer: C

Table 3: Prompts used for collecting closed-questions from Wikipedia pages and collecting responses from open-source models and proprietary models. For 'Collecting responses (proprietary models)', we do not write 'Reason' for all exemplars due to lack of space.
†- Prompts made to GPT-4 Turbo
‡- Prompts made to GPT-3.5 Turbo

SelfCheckGPT with Prompt	Context: <CONTEXT> Sentence: <SENTENCE> Is the sentence supported by the context above? Answer only in Yes or No:
Natural language evaluation using LLM	**convince-factor-wth-answer†** Given the question: <QUESTION> Given the answer: <ANSWER> Given the reason: <REASON> Please score how much the reason convinces you from 1 (not convinced) to 5 (very convinced): **convince-factor-without-answer†** Given the question: <QUESTION> Given the reason: <REASON> Please score how much the reason convinces you from 1 (not convinced) to 5 (very convinced): **factuality‡** Given the source document: <SOURCE> Please score the factuality of the source document from 1 (not factually correct) to 5 (fully factually correct): **information-mismatch‡** Given the source document: <SOURCE> Given the model-generated text: <GENERATED> Please score the amount of information mismatch between source document and model-generated text from 1 (very less mismatch) to 5 (very high mismatch):

Table 4: Prompts for 'SelfCheckGPT with Prompt' hallucination scoring scheme & all modes under leveraging of GPT-3.5 Turbo & GPT-4 Turbo for evaluation various attributes of main response reasoning passages. We use GPT-4 Turbo to verify responses from GPT-4 Turbo itself. Human evaluators are also provided with exactly same prompts.
†- <QUESTION>, <ANSWER> and <REASON> masks are replaced by the current question, main response answer and main response reasoning passage.
‡- <SOURCE> and <GENERATED> are replaced by main response reasoning passage and stochastic response reasoning passages.

models	Percentage invalid main responses	Average percentage invalid stochastic responses
meta-llama-2-7B	1.000	0.558
meta-llama-2-7B-chat	0.152	0.271
meta-llama-2-13B	0.008	0.239
meta-llama-2-13B-chat	0.026	0.099
meta-llama-2-70B	0.136	0.154
meta-llama-2-70B-chat	0.019	0.4
meta-llama-3-8B	0.753	0.45
meta-llama-3-8B-instruct	0.001	0.046
meta-llama-3-70B	0.005	0.219
meta-llama-3-70B-instruct	0.008	0.029
Mistral-7B-Instruct-v0.1	0.693	0.339
Mistral-7B-Instruct-v0.2	0.089	0.138
Mixtral-8x7B-Instruct-v0.1	0.034	0.113
text-davinci-003	0.011	0.011
GPT-3.5 Turbo	0.005	0.011
GPT-4 Turbo	0.000	0.002

Table 5: Percentage invalid responses across all open-source & proprietary models. Low scale models : meta-llama-2-7B & Mistral-7B-Instruct-v0.1 report highest percentage of invalid main responses. GPT models report lowest percentage invalid responses.

Models	H_0 (Null Hypothesis)	H_1 (Alternate Hypothesis)	p-value	degrees of freedom (df)	Result
' ' all-MiniLM-L6-v2	$\mu_{\text{gpt-4-turbo}} = \mu_{\text{gpt-3.5-turbo}}$	$\mu_{\text{gpt-4-turbo}} > \mu_{\text{gpt-3.5-turbo}}$	5.62e-10	979.96	Reject Null Hypothesis
	$\mu_{\text{gpt-4-turbo}} = \mu_{\text{text-davinci-003}}$	$\mu_{\text{gpt-4-turbo}} < \mu_{\text{text-davinci-003}}$	0.0037	979.87	Reject Null Hypothesis
	$\mu_{\text{text-davinci-003}} = \mu_{\text{gpt-3.5-turbo}}$	$\mu_{\text{text-davinci-003}} > \mu_{\text{gpt-3.5-turbo}}$	3.55e-17	985.99	Reject Null Hypothesis
' ' all-mpnet-base-v2	$\mu_{\text{gpt-4-turbo}} = \mu_{\text{gpt-3.5-turbo}}$	$\mu_{\text{gpt-4-turbo}} > \mu_{\text{gpt-3.5-turbo}}$	4.08e-09	981.86	Reject Null Hypothesis
	$\mu_{\text{gpt-4-turbo}} = \mu_{\text{text-davinci-003}}$	$\mu_{\text{gpt-4-turbo}} < \mu_{\text{text-davinci-003}}$	2.58e-05	980.16	Reject Null Hypothesis
	$\mu_{\text{text-davinci-003}} = \mu_{\text{gpt-3.5-turbo}}$	$\mu_{\text{text-davinci-003}} > \mu_{\text{gpt-3.5-turbo}}$	5.28e-21	985.85	Reject Null Hypothesis

Table 6: Welch's t-tests for testing difference in means of hallucination scores given to main response reasoning passages under SelfCheckGPT with BERTScore method. The level of significance for all these tests is 0.05. Note: We assumed the normality of distribution of the hallucination scores for each of the proprietary model and we did not assume anything about their variances.

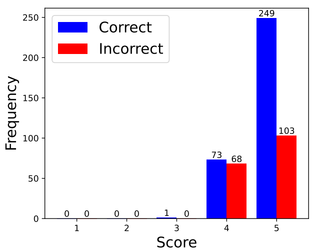
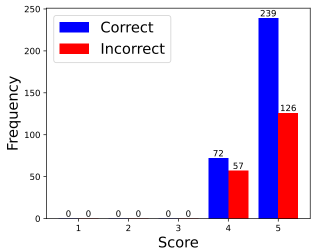
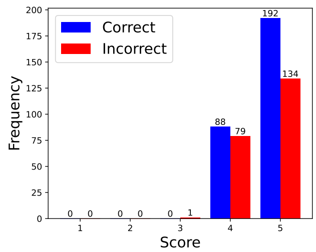
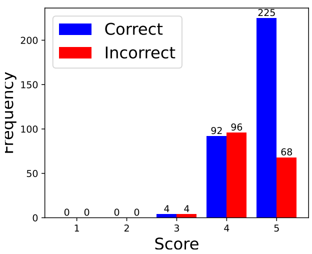
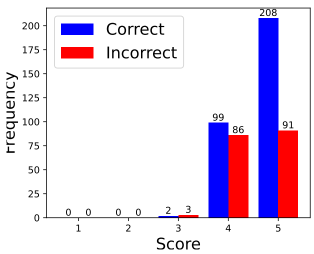
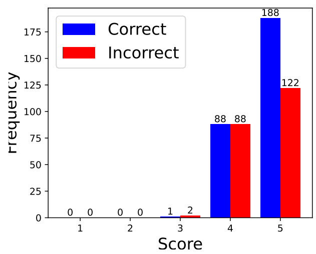
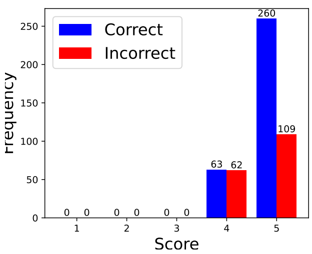
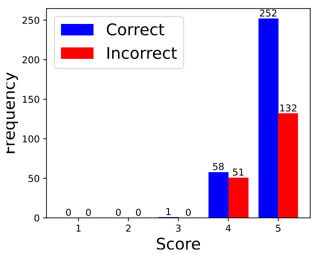
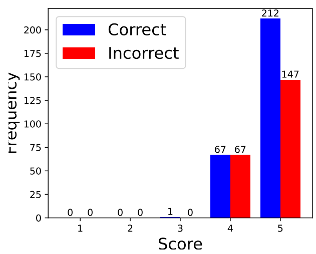
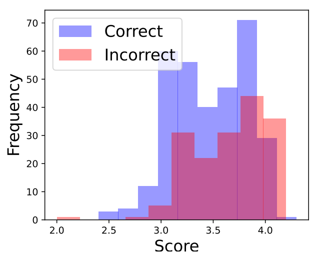
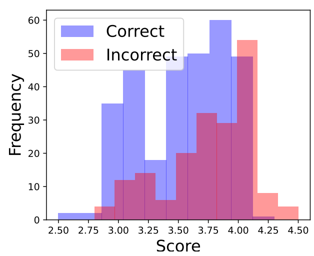
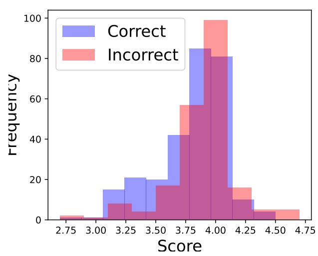
attribute	GPT-4 Turbo	GPT-3.5 Turbo	text-davinci-003
convince-factor-with-answer			
convince-factor-without-answer			
fact-check			
information-mismatch			

Table 7: Verification of the main response reasoning passages generated by all three proprietary models across convincingness (with and without answer), factuality, and information mismatch using GPT-3.5 Turbo as the verifier model.