Prompt Engineering Techniques for Language Model Reasoning Lack Replicability

Laurène Vaugrante *laurene.vaugrante@iris.uni-stuttgart.de*
Interchange Forum for Reflecting on Intelligent Systems
University of Stuttgart

Mathias Niepert
Institute for Artificial Intelligence
University of Stuttgart

Thilo Hagendorff
Interchange Forum for Reflecting on Intelligent Systems
University of Stuttgart

Reviewed on OpenReview: *https://openreview.net/forum?id=bgjR5bM44u*

Abstract

As large language models (LLMs) are integrated into everyday applications, research into prompt engineering techniques (PET) to improve these models' behavior has surged. However, clear methodological guidelines for evaluating these techniques are lacking. This raises concerns about the replicability and generalizability of the prompt engineering techniques' benefits. We support our concerns with a series of replication experiments focused on zero-shot prompt engineering techniques purported to influence reasoning abilities in LLMs. We tested GPT-3.5, GPT-4o, Gemini 1.5 Pro, Claude 3 Opus, Llama 3, Vicuna, and BLOOM on the chain-of-thought, Sandbagging, EmotionPrompting, Re-Reading, Rephrase-and-Respond (RaR), and ExpertPrompting prompt engineering techniques. We applied them on manually double-checked subsets of reasoning benchmarks including CommonsenseQA, CRT, NumGLUE, ScienceQA, and StrategyQA. Our findings reveal a general lack of statistically significant differences across nearly all techniques tested, highlighting, among others, several methodological weaknesses in previous research. To counter these issues, we propose recommendations for establishing sound benchmarks, and designing rigorous experimental frameworks to ensure accurate and reliable assessments of model outputs.

1 Introduction

The field of generative artificial intelligence has considerably evolved in only a few years. In particular, large language models (LLMs) have witnessed an unprecedented surge in popularity with the release of ChatGPT (OpenAI, 2022), which became the most rapidly adopted internet application in history. LLMs possess advanced natural language processing capabilities which demonstrate a broad range of downstream applications, ranging from casual conversations to complex problem-solving (Minaee et al., 2024; Zhou et al., 2020). Given the fast growing range of applications (Guo et al., 2024) plus their respective risks for AI alignment (Ji et al., 2024), fairness (Hao et al., 2023), and safety (Amodei et al., 2016; Hagendorff, 2024; Vaugrante et al., 2025; Weidinger et al., 2023), it is paramount to evaluate behavioral and reasoning patterns these models exhibit (Binz & Schulz, 2023; Gao et al., 2025; Wang et al., 2024a). This created the need for new research fields, and fostered a variety of different approaches to investigate different types of LLM behavior, including emergent abilities and prompt engineering strategies (Chang et al., 2023). A substantial part of this research relies on prompt engineering techniques, which are designed to steer LLMs toward desired responses without modifying their internal structure (Liu et al., 2021). However, prior work has demonstrated that some of these techniques fail to replicate in newer, reasoning-oriented models, particularly in software-engineering contexts (Wang et al., 2024b). We hypothesize that such replication challenges may extend beyond reasoning models and occur even with instruction-following LLMs across a broader range of reasoning tasks, indicating a deeper replication problem within prompt-engineering research. To investigate this, we conduct experiments attempting to conceptually replicate studies investigating zero-shot prompt engineering techniques that are believed to enhance reasoning in LLMs. Our findings reveal that these techniques often fail to produce consistent improvements. Based on our findings, we propose specific recommendations and tools for developing better methodologies when evaluating LLM behavior. This includes establishing sound benchmarks, designing robust experimental frameworks adapted to the LLMs used, and implementing accurate evaluations of model outputs.

2 Methods

2.1 Prompt Engineering Techniques

For our experiments, we tried to replicate single-pass, zero-shot prompt engineering techniques that were demonstrated to alter reasoning performances in LLMs in previous studies:

- **Zero-shot chain-of-thought Prompting** (Kojima et al., 2022): This method claims that adopting a step-by-step reasoning approach in LLMs enhances overall reasoning accuracy.
- **Sandbagging** (Perez et al., 2022): Sandbagging showcases that LLMs have a tendency to repeat back a dialog user's preferred response and mirror them when solving tasks.
- **EmotionPrompting** (Li et al., 2023): This technique consists in adding emotional stimuli, such as "This is very important to my career", in order to enhance the performance.
- **Re-Reading** (Xu et al., 2024): This method consists in repeating the task twice to enhance the reasoning accuracy.
- **Rephrase-and-Respond** (Deng et al., 2024): This approach involves rephrasing the given task as a query before providing a response, thereby improving accuracy.
- **ExpertPrompting** (Xu et al., 2025): This technique claims to enhance the LLM accuracy when setting the LLM in an expert role.

With this selection, we aim at covering the majority of prompt engineering techniques that represent zero-shot, single-pass methods which are most likely to be adopted by a wide range of users in real-world LLM settings. Furthermore, these techniques simplify the experimental design and minimize potential sources of error, which would be present in more complex settings. We deliberately excluded methods such as ensembling, self-criticism, least-to-most prompting, tree-of-thoughts, etc., as these involve more complex setups or prompt chains (Schulhoff et al., 2024; Yao et al., 2023) which we consider unlikely to be utilized by typical LLM users. As prompt engineering techniques can be applied to various applications (Liu et al., 2021) like improving text quality, enhancing the answer's relevance or controlling the output formatting, we chose to focus on prompt engineering techniques used to enhance reasoning abilities and hence accuracy of LLMs when prompted with complex tasks.

2.2 Benchmark Selection

To replicate the claimed impact of the selected prompt engineering techniques on LLM reasoning abilities, we selected five different benchmarks, each measuring a different type of reasoning: CommonsenseQA (Talmor et al., 2019), CRT (Hagendorff et al., 2023), NumGLUE (Mishra et al., 2022), ScienceQA (Lu et al., 2022) and StrategyQA (Geva et al., 2021). In accordance with a growing body of research (Gema et al., 2024; Goetze & Abramson, 2021) we noticed a low quality of many benchmark items, meaning incorrect or ambiguous questions, formatting flaws, or factual errors in the response choices. Therefore, we chose to hand-pick (through rule-based filtering and manual checks) 150 faultless questions out of a random sample of 200 questions per benchmark, with a total of $n = 750$, preferring accuracy over large sample sizes. The tasks were either open-ended, Boolean, or multiple-choice questions.

2.3 Experiments

We first measured the accuracy of LLMs in a base test using unmodified tasks. We then applied the prompt engineering techniques outlined in the studies mentioned above by incorporating the necessary pre- or suffixes to each task. We used the same prompts described in these studies when available, and generated new ones based on the prompt descriptions when they were not. When the studies used several pre- or suffixes as a basis to their claim, such as in the EmotionPrompting study where 11 different emotional stimuli were used, we randomly selected one of them for each task using a seed.

2.4 LLM selection

We compared the performance of five different LLMs, in particular OpenAI's GPT-3.5 (gpt-3.5-turbo) (OpenAI, 2022) and GPT-4o (gpt-4o-2024-05-13) (OpenAI, 2023), Google's Gemini 1.5 Pro (gemini-1.5-pro-001) (Team et al., 2024), Anthropic's Claude 3 Opus (claude-3-opus-20240229) (Anthropic, 2024), Google's Gemini 1.5 Pro (gemini-1.5-pro-001) (Team et al., 2024), and Meta's Llama 3, with both 8B and 70B versions (Meta-Llama-3-8B-Instruct and Meta-Llama-3-70B-Instruct) (34). As the selected studies also used some models from earlier generations, we have also attempted to measure the performance for Vicuna 13B v1.5 (Chiang et al., 2023) as well as BLOOM 176B (Workshop et al., 2023), but the obtained results were deemed unusable due to the models' inability to generate coherent outputs, their tendency to produce meaningless loops, repeated fragments of the input, and other issues, as detailed in Appendix A.

2.5 Output classification

To facilitate the LLM output classification process without restricting the reasoning behavior during the LLMs' prompt completions, we added an instruction to write the final answer after a specific string, namely "####", to each benchmark task, as indicated in the literature (Cobbe et al., 2021; Nezhurina et al., 2024). We then assessed the LLM outputs following "####" by combining string matching methods, LLM-based evaluations with GPT-4o, as well as manual double-checks (see Appendix B). Considering that the behavior of LLMs might exhibit variations over time (Chen et al., 2024a), we report the timeframe of the experiments. They spanned from June 6th, 2024, to June 17th, 2024, except for the Vicuna 13B and BLOOM experiments, which spanned from December 4th to December 14th, 2024. For all experiments, LLM temperature parameters were set to 0, or 0.00001 when 0 was not permitted.

2.6 Study focus

This study focuses on *replication* rather than on *reproducibility*. According to Peng (2011), replication involves collecting and analyzing new data to replicate the findings of a previously conducted study, whereas reproducibility entails reanalyzing the original data to verify the results. Our hypothesis when replicating the previous experiments was that the claimed performance improvements are not replicable and hence the claims about the prompt engineering techniques are not *generalizable*. We neither use the exact same selections of benchmarks nor models as in the original studies but vary the experimental setups slightly. In detail, this means that we still test foundation text-to-text models, use reasoning benchmarks, and use prompt engineering techniques either verbatim (chain-of–thought, Re-Reading, EmotionPrompting, Rephrase-and-Respond), or, if necessary, adapted in alignment with their original methodology to suit our benchmarks (Sandbagging, ExpertPrompting,), which should theoretically increase their utility. However, it is important to note that our approach to classifying LLM responses likely differs substantially from the original studies, as many of them lack descriptions of their chosen methods, further underscoring the nature of our work as a replication study rather than a reproduction.

2.7 Statistics

All statistical analyses were performed using Python (version 3.11.4). The SciPy library (version 1.13.1) was used for statistical computations, while visualizations were created with Matplotlib (version 3.7.1) and Seaborn (version 0.12.2). We applied chi-squared (χ^2) tests to assess the statistical significance of accuracy differences between baseline and modified prompts. Rounded P-values are reported for each test. 95% confidence intervals (CIs) were calculated and included in the result visualizations. Performance variability across models and benchmarks was accounted for, and results were reported per model, benchmark, and prompt engineering technique.

3 Results

3.1 Chain-of-thought prompting

Chain-of-thought prompting involves decomposing a given task and solving each step before outputting the final answer, by presenting the LLM with an example of a task and its expected decomposed output. In the original study establishing this method, Wei et al. (2023) tested five LLMs over three reasoning, categories including arithmetic reasoning, commonsense reasoning, and symbolic reasoning, harnessing 12 different benchmarks. The authors claim a good robustness of this method, with several different annotators. While they reported variance in the average performance, it was consistently superior to the performance with the base evaluation, with a reported average improvement of 39.91% (Wei et al., 2023). A subsequent study then claimed that a zero-shot chain-of-thought prompting strategy sufficed to elicit similar improvements (Kojima et al., 2022). Instead of presenting, before each task, an example enabling chain-of-thought reasoning, they simply suffix tasks with "Let's think step by step". They tested a larger sample of 17 LLMs on various reasoning categories, utilizing 12 benchmarks akin to the previous paper. They obtained an averaged 35.93% improvement in accuracy for zero-shot chain-of-thought reasoning across all benchmarks and models (Kojima et al., 2022). We tried to replicate these findings with our set of reasoning benchmarks. However, despite the impressive results from the original studies, we observed that there was no significant improvement (see Figure 1): with the exact same task suffix as in the original study, we could not observe any significant difference across all benchmarks. With results from all models combined, the maximal positive impact of chain-of-thought reasoning is with NumGLUE where there is a 2.78% accuracy difference between the base and the chain-of-thought prompt (see Appendix C), which is not significant given the total number of tasks ($\chi^2 = 1.78, p = 0.18$). These numbers remain similar throughout each LLM evaluated, with an overall average improvement of 0% for the chain-of-thought reasoning ($\chi^2 = 0.06, p = 0.8$), as seen in Appendix C. The largest observed positive impact of chain-of-thought reasoning is for Llama 3-70B tasked by CommonsenseQA, with an observed 8.67% improvement ($\chi^2 = 2.19, p = 0.14$) (see Appendix C). However, the highest overall difference is an 11.33% accuracy decrease ($\chi^2 = 4.47, p < .05$) (see Appendix C) with chain-of-thought reasoning applied on Llama 3-70B with StrategyQA. While the latest models seem to implement chain-of-thought reasoning by default, meaning without being specifically prompted to, these results hold even for previous models such as GPT-3.5, which often do not. We compared the average response length of each LLM when chain-of-thought reasoning is explicitly requested, compared to when it is not, as shown in Appendix D. Even when the base experiments do not demonstrate verbose prompt completions and the chain-of-thought prompting does, the performance results are not impacted in a significant manner, which stands contrary to what the literature suggests (Jin et al., 2024). For instance, GPT-4o had an average difference of response lengths of 531 characters for the base test vs. 931 characters for the chain-of-thought prompting, but just a 0.01% accuracy difference, suggesting that simply increasing the length of prompt completions does not enhance accuracy beyond a certain point.

3.2 Sandbagging

Perez et al. (2022) demonstrate sycophancy, which is an LLM's tendency to output answers that users tend to prefer. The researchers evaluated several aspects of sycophancy, including a "sandbagging" capability, which suggests that a model could underperform when a user is deemed incapable to solve or verify a given task. They underpin this hypothesis by adding user biographies before reasoning tasks from TruthfulQA (Lin et al., 2022), with "very educated" users as opposed to "very uneducated" users. They imply a significant difference between these two categories, claiming that sandbagging causes LLMs to output incorrect answers when human users are perceived as unable to answer correctly themselves (Perez et al., 2022). We conceptually replicate this experiment using our selected models by prefixing our selected reasoning tasks with both "very educated" and "very uneducated" user biographies (see Appendix E). We observe no significant difference over all benchmarks when comparing the highly educated ($\chi^2 = 1.64, p = 0.20$) or poorly educated ($\chi^2 = 1.24, p = 0.27$) user prompts to the base results (see Figure 1 and Appendix C), with an average accuracy decrease of 1% for both cases (see Appendix C). We likewise observe no significant difference when comparing the highly educated to the poorly educated user prompt results, and frequently observe that the "poor education" prefixed tasks have an even better performance than the "high education" ones (average accuracy improvement of 0.1% for "poor education"). Once again, we fail to replicate the sandbagging phenomenon when utilizing our experimental setup.

3.3 EmotionPrompting

EmotionPrompting, presented by Li et al. (2023), augments a task with emotional cues such as "You'd better be sure" or "This is very important to my career" to enhance problem-solving abilities in LLMs. In the original study, Li et al. (2023) augmented tasks with 11 variations of emotional stimuli and tested six LLMs including ChatGPT and GPT-4. They sourced their tasks from three different benchmark categories, notably using tasks from BIG-Bench (Srivastava et al., 2022). They claim to obtain a "relative performance improvement of 115%" (Li et al., 2023) with their method, arguing that adding an emotional component improves the capabilities of LLMs. However, despite the improvement that was strongly implied throughout the original study by raising claims like "EmotionPrompt makes it easy to boost the performance of LLMs" (Li et al., 2023), the numerical values communicated in the study itself do not coincide with these claims. Instead of communicating the average improvement of the enhanced prompts over the regular prompts, they focused on improvements when cherry-picking the most performant emotional cue. Based on their reported results, we calculated an average relative performance improvement of 4.42% on BIG-Bench tasks, and a 2.58% relative performance improvement across all benchmarks, when choosing the average performance of all emotional stimuli. Despite identifying this shortcoming in the original study at this early stage, we nevertheless replicated the experiments with our selected tasks and models. We applied the same emotional suffixes as in the original study, apart from "Are you sure?", as LLMs tend to reply to this question, as opposed to solving the given tasks. Similarly to Li et al. (2023)'s findings, but contrary to their claims, we observed that there was no significant improvement, across every single model and benchmark (see Figure 1). The maximal positive improvement measured is non-significant with an 8.7% difference ($\chi^2 = 1.94, p = 0.16$) (see Appendix C), using Llama 3-8B on CommonsenseQA. Overall, we observe an insignificant performance increase of 1% when applying EmotionPrompting ($\chi^2 = 0.11, p = 0.74$) (see Appendix C).

3.4 Re-Reading

Re-Reading, introduced by Xu et al. (2024), consists in repeating the task verbatim before having the model answer. They compared the baseline performance with the Re-Reading performance, as well as the performance in both conditions when additionally suffixing every task with a chain-of-thought eliciting prompt. The researchers tested GPT-3 (text-davinci-003) (Brown et al., 2020), GPT-3.5, Llama-2-13B and Llama-2-70B (Touvron et al., 2023), in order to have both models with and without instruction fine-tuning. They used a total of 112 arithmetic, common sense, and symbolic reasoning tasks sourced from various datasets with GPT-3 and GPT-3.5, for which they obtained an average gain of 2.7% in accuracy, and 2.9% with the inclusion of chain-of-thought reasoning. For Llama 2-13B and Llama 2-70B, they used a different set of benchmarks comprising only arithmetic reasoning tasks, with an average gain of 2.5% in accuracy (2.7% with chain-of-thought reasoning). We replicate the Re-Reading experiments on our selected tasks and models. For this study, we observe a significant improvement for Llama 3-8B ($\chi^2 = 13.13, p < .05$) and Llama 3-70B ($\chi^2 = 19.4, p < .05$) exclusively (see Appendix C and Figure 1). The maximal improvement across all benchmarks for the other models is of 2%, for Claude 3 Opus ($\chi^2 = 1.27, p = 0.26$). Therefore, Re-Reading seems replicable for the Llama 3 models only, which highlights the importance of implementing tests on a variety of models. However, the initial study indicated that Re-Reading was effective on GPT models, notably GPT-3.5, that we also tested with different outcomes. Therefore, we only managed to partially replicate the results.

3.5 Rephrase-and-Respond

Rephrase-and-Respond, inspired by the communication technique of rephrasing to enhance clarity, involves instructing an LLM to first rephrase a task and then address it within the same prompt. Deng et al. (2024) demonstrate that simply adding a directive such as "Rephrase and expand the question, and respond" can significantly enhance LLM accuracy. They evaluated GPT-3.5, GPT-4 and Llama 2 across seven different reasoning benchmarks, such as CommonsenseQA and Knowledge Classification (Allen-Zhu & Li, 2024), and observed an average improvement of 17.80% across all tested models. We replicated this experiment by incorporating the suggested rephrasing directive to reformulate questions across our full dataset. We then applied a tailored algorithm to classify the answers, ensuring it could accommodate these new types of responses (see Appendix B). As for the Re-Reading experiment, we only observe a significant positive effect for Llama 3-8B ($\chi^2 = 10.49, p < .05$) and Llama 3-70B ($\chi^2 = 18.63, p < .05$), with the StrategyQA benchmark. Once again, Rephrase-and-Respond seems to only show a significant positive improvement on a specific benchmark-model combination, which shows only a partial replication of the results (see Figure 1 and Appendix C).

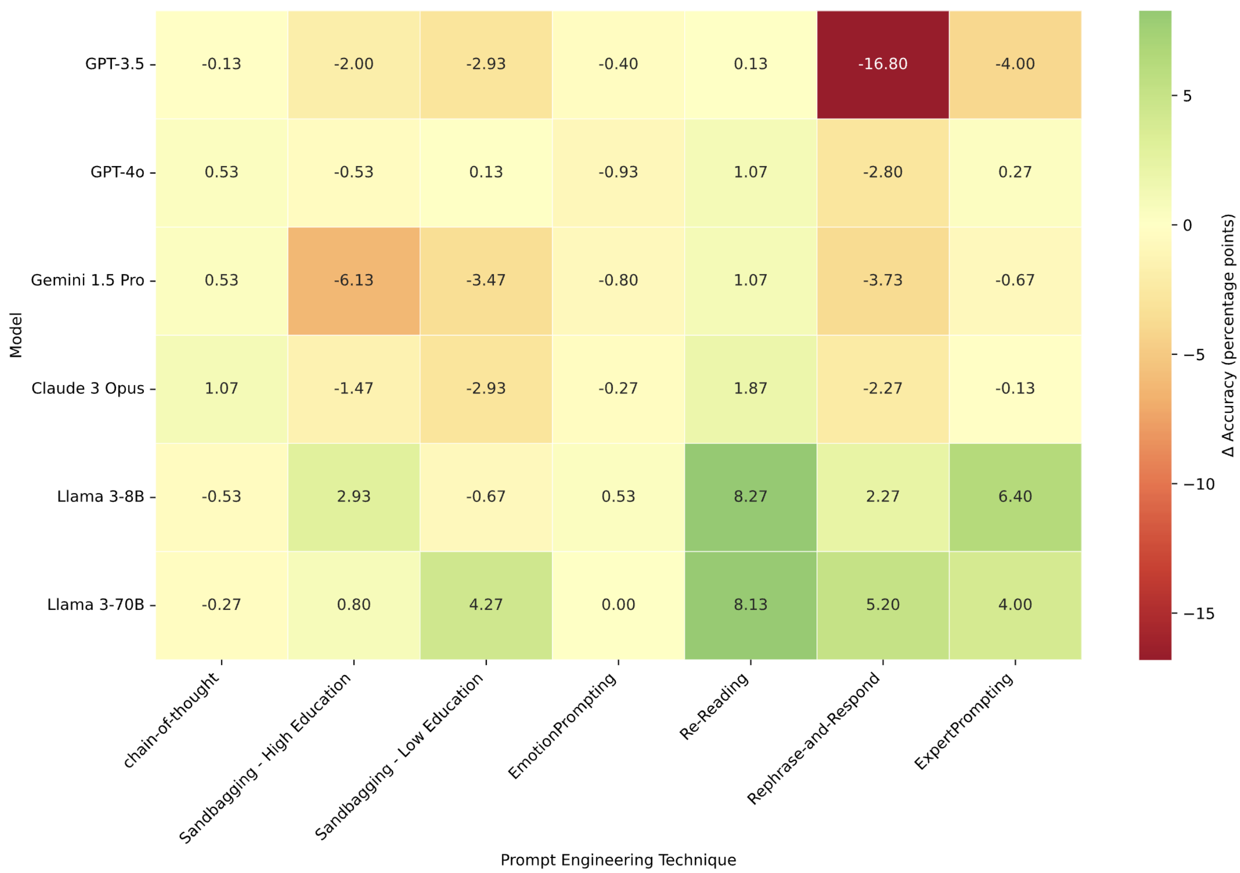

Figure 1: Accuracy comparisons between the base tests without any prompt modification and the augmented prompts across all LLMs.

3.6 ExpertPrompting

ExpertPrompting consists of giving LLMs an instruction to impersonate someone with high expertise on the task subject while completing a task. This method presented by Xu et al. (2025) has been greatly popularized; it is now recommended in LLM documentations for enhanced LLM accuracy and improved focus on adhering to the task's requirement. Xu et al. (2025) evaluated the response quality of ExpertPrompting, assessing aspects like accuracy, helpfulness, or relevance. They evaluated GPT-4 responses with and without ExpertPrompting, which, in the case of the former, possessed a reported higher answer quality 48.5% of the time (Xu et al., 2025). In our experiments, we measure, as with the previous prompt engineering techniques, the accuracy of the ExpertPrompting technique using our set of reasoning benchmarks and LLMs. We observe no significant improvement across all benchmarks ($\chi^2 = 1.57, p = 0.21$) (see Figure 1 and Appendix C), with an average improvement of only 1% (see Appendix C).

4 Identified Issues and Recommendations

Given the identified lack of replicability across various studies, we deem crucial to address the underlying issues contributing to these replication problems in prompt engineering. We have identified specific problems associated with each cited prompt engineering method, which may hint at a broader replication problem in LLM behavior research. In the following, we provide recommendations aimed at mitigating replication problems in prompt engineering and related fields, based on our analysis of the selected studies: (1) benchmark adequacy, (2) model variability and model-benchmark compatibility, and (3) output evaluation and reliability. It is important to note that while an absolute elimination of replication problems is unlikely, our recommendations aim to minimize such challenges effectively.

4.1 Benchmark adequacy

4.1.1 Benchmark Quality

In several of the studies examined, we noted major issues regarding the benchmarks used to assess LLM performances. Many benchmark tasks are flawed: they lack proper grammar, contain spelling or punctuation errors (e.g., missing question marks in multiple-choice questions), are semantically nonsensical, incomplete, or factually incorrect. For instance, for CommonsenseQA (used in the chain-of-thought, Re-Reading, and Rephrase-and-Respond studies), 10.9% of questions presented punctuation issues easily verifiable with a simple code. Such flaws are not isolated incidents: they reflect a broader pattern of benchmark instability that has been widely discussed in the literature (Gema et al., 2024; Goetze & Abramson, 2021). Label errors can destabilize benchmark reliability, and even small proportions of mislabeled test data can invert model rankings: a phenomenon they call the capacity–performance paradox (Northcutt et al., 2021). This effect implies that benchmark noise may disproportionately penalize higher-capacity models, creating misleading impressions of their reasoning ability. Similarly, researchers have revealed significant ground-truth errors in the Massive Multitask Language Understanding (MMLU) benchmark, one of the most widely used evaluation suites for LLMs (Gema et al., 2024). Their re-annotated version, MMLU-Redux, shows that previously reported model hierarchies often vanish once test data are corrected. Likewise, prior work has exposed flaws in widely used reasoning benchmarks, revealing that evaluation setups often reward superficial lexical alignment rather than genuine reasoning (Mousavi et al., 2025). Together, these findings align with our own observations: large benchmark datasets frequently contain nonsensical, ambiguous, or incorrect items that can distort results.

To mitigate benchmark-induced replication failures, we recommend that researchers rigorously validate benchmark datasets before use. Automated scripts can detect obvious errors (e.g., typos, missing punctuation, malformed options), but human verification remains essential for identifying semantic inconsistencies. Following research recommendations (Northcutt et al., 2021; Cao et al., 2025), benchmark curation should be treated as a core methodological step rather than an auxiliary task. We recommend mentioning the cleaning procedure and error rates, and encourage the adoption of transparent guidelines such as How2Bench (Cao et al., 2025). Where possible, studies should prioritize quality over quantity, using smaller but validated subsets rather than relying on noisy large-scale test sets. This approach minimizes variance, prevents benchmark-driven illusions of improvement, and enhances comparability across studies. Finally, we advocate for the publication of "benchmark versioning" and reproducible evaluation code, so that future research can easily verify whether results stem from model performance or from artifacts in the evaluation data.

4.1.2 Benchmark Coherence and Cross-Study Comparability

A second major source of replication instability arises from inconsistent benchmark selection within and across studies. In several prompt-engineering papers, models are evaluated on distinct benchmark sets within the same experiment. For example, in the Re-Reading study (Xu et al., 2024), GPT models were tested on three reasoning benchmarks, whereas Llama 2 models were evaluated on only one. In the sandbagging study (Perez et al., 2022), researchers evaluate differences in model accuracy when answering questions on the TruthfulQA dataset (Lin et al., 2022), which measures whether a language model is truthful in generating answers to questions, so whether the facts mentioned in the answer are correct rather than assessing whether they answered the task correctly. Such inconsistency complicates direct comparison of model performance and raises questions about the rationale behind benchmark choices, potentially leading to benchmark cherry-picking that favors results supporting a given hypothesis. Beyond within-study inconsistencies, there is little coherence in benchmark use across related works. In our replication, we applied the same benchmark suite across all experiments to ensure comparability. In contrast, the six studies we examined drew on widely different datasets, some domain-restricted, others deterministic in nature. For instance, the chain-of-thought study evaluated reasoning-centric tasks, while the EmotionPrompting paper used tasks from Instruction Induction (Honovich et al., 2022) and BIG-Bench (Srivastava et al., 2022), such as rhyming or pluralization, which makes results hard to compare as these measure different underlying abilities. Similarly, variations within the process of administering benchmark tasks (notably zero-shot prompting versus few-shot prompting) impact the reasoning process of LLMs and therefore the outcomes. These issues mirror findings by researchers who conducted a large-scale analysis of benchmark agreement and showed that minor methodological differences, such as prompt format, evaluation metric, or dataset choice, can drastically alter perceived model performance (Perlitz et al., 2024). Finally, it is essential to select benchmarks that are conceptually and structurally aligned with the research question. Indeed, when testing prompt techniques claiming to alter reasoning performance, given or implied response instructions in the tasks can interfere with output accuracy by restricting the response length and therefore its ability to generate more detailed responses, for instance in multiple-choice settings.

Table 1: Benchmark-related replication issues, mitigation techniques, and tools

Issue	Mitigation	Tools
Flawed benchmark tasks: may inflate or deflate perceived performance and favor some models	• Include benchmark curation as a core methodological step • Prioritize quality over quantity • Report quality-check methods and error rates	• Automated validation scripts • Guidelines such as How2Bench (Cao et al., 2025)
Inconsistent benchmark selection: complicates cross-study comparison	• Report exact benchmarks and what they measure • Adopt standardized benchmark-agreement protocols	• BenchBench framework (Perlitz et al., 2024) • Benchmark agreement protocol
Benchmark–objective misalignment: may distort perceived performance by assessing unrelated capabilities	• Ensure coherence between benchmarks and research objectives • Document rationale for benchmark choice	• BenchBench framework (Perlitz et al., 2024) • Benchmark agreement protocol

We recommend preferring standardized benchmarks across studies in the same field, to ensure that benchmarks are closely aligned with the research objectives. We would advise researchers to ensure that benchmark domains match the specific research objective. We further recommend adopting benchmark-agreement protocols such as those implemented in BenchBench (Perlitz et al., 2024), which promote standardized evaluation and transparent documentation of benchmark configurations. Establishing and reporting such consistency will reduce variance, mitigate cherry-picking concerns, and make replication efforts more meaningful.

4.2 Model Variability and Model-Benchmark Compatibility

4.2.1 Model Variability

In some cases such as in the chain-of-thought prompting study (Kojima et al., 2022), the lack of replicability is linked to the models' type and generation. Our results reveal a divergence between GPT-3.5 and GPT-4o: the former shows mild improvement with chain-of-thought prompting, whereas the latter displays no significant change. Although neither result reaches statistical significance, this difference suggests that for earlier-generation models (e.g., GPT-3), used in the replicated study along with other models of that generation, chain-of-thought prompting might have successfully improved LLM accuracy. This aligns with the system cards for recent models, which explicitly warn that techniques like chain-of-thought reasoning may not improve performance and can even impair it, advising caution in their use with these models (OpenAI, 2024). Similarly, we observed a significant improvement for some benchmarks with the Re-Reading and Rephrase-and-Respond prompts, for the Llama 3 models exclusively; if we look at the other models separately, the results are vastly non-significant (see Appendix C). This reinforces the claim that similar experiments may have a considerably different impact depending on the models used. Moreover, even when conducting replication experiments using the same models as in the original study, the opacity surrounding model updates and developer prompt modifications in terms of date and type of update (Chen et al., 2024a) renders study replications difficult. Furthermore, the efficacy of prompt engineering depends strongly on the model class; whether the model is inherently reasoning-oriented or relies on surface-level pattern completion. This dynamic is consistent with prior work demonstrating that prompt engineering substantially benefits non-reasoning LLMs but provides marginal or inconsistent improvements for reasoning-tuned ones such as o1-mini (Wang et al., 2024b).

Therefore, it is essential to use a variety of models when testing a hypothesis, or to at least mention the limited scope of the study when fewer models are used, as an effort to prevent a generalization that may be incorrect. In addition to that, some earlier-generation models may not be well-suited for such question-answering experiments requiring reasoning abilities. Specifically, we tested Vicuna 13B and BLOOM on our selected tasks (Appendix A), but found the results difficult to interpret, as the accuracy for both the base questions and the prompt-engineered questions was too low to draw reliable insights into the models' behavior. The studies on EmotionPrompting, Rephrase-and-Respond and ExpertPrompting used Vicuna or BLOOM nonetheless. We recommend consistently clarifying the methodology used for each model, as we were unable to understand how these studies achieved viable results with these models. We recommend that future studies:

1. Explicitly classify models by reasoning capacity when reporting results, distinguishing instruction-following models from reasoning-optimized ones.
2. Document model version and release date to ensure comparability over time, as minor provider-side updates can significantly alter performance.

Clarifying these methodological details will help disentangle true performance differences from artifacts introduced by model selection, and will prevent overgeneralization from results obtained on older, non-reasoning LLMs to newer architectures that already internalize these behaviors.

4.2.2 Model-benchmark Compatibility

As large language models evolve, benchmark saturation and data contamination have become central challenges in evaluating prompt-engineering methods. As newer models improve in baseline reasoning performance, many existing benchmarks become effectively saturated, leaving little room for prompt engineering to demonstrate measurable gains. For example, tasks that once challenged early-generation models are often solved at near-ceiling accuracy by later models, as we can see with the CRT benchmark (Hagendorff et al., 2023). This saturation reduces statistical sensitivity and can obscure subtle effects introduced by prompting techniques. It is therefore essential to continuously recalibrate benchmark difficulty to match evolving model capabilities. Similarly, many zero-shot and few-shot benchmarks suffer from task leakage (Li & Flanigan, 2024; Wu et al., 2025; Zhang et al., 2025): models perform substantially better on datasets predating their training cutoff than on unseen ones, implying widespread contamination. These findings highlight that contamination can make both baseline and prompt-engineered performances appear inflated, thereby diminishing the apparent impact of prompting interventions.

Leakage-free benchmarks such as RandomCalculation (Wu et al., 2025) can be used to isolate genuine reasoning improvements. Likewise, researchers introduce a reasoning-driven synthesis framework that generates new research-level QA pairs automatically verified for correctness (Zhang et al., 2025). This method is effective for mitigating contamination and memorization effects. Moreover, we recommend using canary strings when publishing benchmark data to help detect and quantify potential data leakage in future models trained on that data. By embedding unique, traceable strings within benchmark datasets, researchers can later probe models to see whether these sequences are reproduced, thus revealing unintended memorization. This proactive approach further enhances the reliability of evaluation results.Finally, prompt effectiveness depends on both model architecture and task characteristics (Chen et al., 2024b). For an optimal performance, different prompts should be adapted to different LLMs, which may explain some differences observed in this study, especially when the original prompt engineering studies were applied on a small sample of LLMs. A "one-size-fits-all" benchmark evaluation approach can no longer capture the true impact of prompt-engineering methods, without comparing the exact same models. Other than expanding the number and variety of LLMs tested, a possible solution would be to adapt these prompt engineering techniques to each LLM using model-adaptive prompt optimizers (Chen et al., 2024b) to calibrate the prompts to the models.

We recommend that future studies:

1. Adjust benchmark difficulty to prevent saturation: when newer reasoning models reach near-ceiling performance, evaluations should shift toward more complex or reasoning-intensive tasks.
2. Mitigate contamination by verifying dataset release dates and removing items that predate model training cutoffs.
3. Employ model-adaptive evaluation strategies that align prompts and benchmarks with each model's capabilities in question, reducing misleading cross-model comparisons.
4. Report benchmark provenance and contamination checks as part of all experimental documentation, alongside model version and date, to promote transparency and interpretability.
5. Broaden model coverage to strengthen the generalizability of findings.

Ultimately, as LLMs integrate reasoning processes internally, older benchmarks cease to serve as meaningful discriminators of model capability or prompt-engineering effectiveness. Reliable evaluation now requires not only harder benchmarks such as Humanity's Last Exam (Phan et al., 2025), but also methodologies that explicitly account for model–benchmark interactions: how specific benchmarks align with, advantage, or contaminate particular models, thereby shaping observed results.

4.3 Output Evaluation and Reliability

4.3.1 Evaluation Reliability

For evaluations of LLMs, it is key to ensure the accuracy of the LLM output classifications. We have attempted to replicate a large number of verification techniques presented in other studies. However, when checking the accuracy of these techniques, we discovered that a significant number of them had shortcomings. For example, functions based solely on Regex rules were generally too vague, leading to flawed classifications.

Table 2: Model-related replication issues, mitigation techniques, and tools

Issue	Mitigation	Relevant Tools
Model variability and opacity: performance differences across model generations and families	• Classify models by reasoning capacity (reasoning vs. non-reasoning) • Document exact model version	
Benchmark saturation and contamination: newer models reach near-ceiling accuracy or show inflated results for pre-cutoff datasets	• Recalibrate benchmark difficulty using reasoning-intensive tasks • Verify dataset release dates against model training cutoffs • Use leakage-free datasets • Check for canary strings	• Leakage-free datasets such as RandomCalculation (Wu et al., 2025) • Beyond Memorization synthesis framework (Zhang et al., 2025) • Harder benchmarks such as Humanity's Last Exam (Phan et al., 2025)
Model–benchmark incompatibility: some benchmarks advantage or disadvantage certain model types, skewing generalization	• Conduct cross-model evaluations on identical benchmarks	

Other metrics, such as the F1 word overlap score, do not work effectively, as they would classify correct LLM outputs as incorrect, simply because the token length differed too much from the ground truth. Similar issues have been found with metrics like BLEU, ROUGE or METEOR (Ng & Abrecht, 2015; Liu et al., 2017). Moreover, studies often rely on using LLM-as-a-judge to classify LLM outputs (Pan et al., 2023), such as the ExpertPrompting study (Li et al., 2023). Exclusively relying on LLM-based evaluation can introduce flaws. When applying the original or even enhanced versions of their LLM output classification prompts, we observed multiple issues that compromised evaluation reliability. For instance, the judge would produce inconsistent evaluations when asked multiple times to assess the same output. It was also influenced by the task when classifying responses, and the verification prompt needed to be task-specific and highly precise. Prior research has also documented LLM-as-a-judge biases that may further distort classifications, such as the self-enhancement bias, a tendency for an LLM judge to rate outputs from its own model family more favorably, or the verbosity bias, where longer responses are rated more positively (Zheng et al., 2023). As some prompting techniques, such as chain-of-thought prompting, systematically increase response length (Appendix D), verbosity bias may artificially inflate measured accuracy when using LLM judges. Given the time-consuming aspect of manually double-checking classifications, we also suspect that this is done very rarely. Furthermore, among the papers we replicated, one did not include any details about the verification process (Li et al., 2023); it goes without saying that any verification process should be clearly reported in each study, to make the study replication feasible. We therefore recommend developing more precise and task-specific verification methods, and ensuring thorough documentation of these processes in all studies to facilitate accurate replication and validation of results. We recommend avoiding reliance solely on LLM-based evaluation; if alternative evaluation strategies are not feasible, then we would recommend running multiple iterations per prompt and aggregating the results to mitigate potential misjudgments. We also recommend that creators of new benchmarks provide a standardized verification process, encouraging all users to apply the same verification criteria.

4.3.2 Evaluation Metrics

The choice of evaluation metric is as critical as the verification method itself. Different metrics capture fundamentally distinct dimensions of model behavior, and results are not directly comparable across them. For instance, accuracy reflects factual correctness, whereas output quality captures fluency, coherence, or stylistic adequacy. Similarly, pairwise comparison approaches - where an LLM or human judge selects the "better" of two outputs - measure relative preference, not absolute performance. Treating these heterogeneous measures as equivalent can therefore produce misleading conclusions about the effectiveness of a prompting technique. For example, in our study, the original ExpertPrompting study focused on output quality rather than accuracy. We replicated their comparative evaluation approach using our own benchmarks. To ensure an unrestricted quality assessment, we removed formatting instructions from our prompts and evaluated complete LLM outputs. Our results showed higher quality scores for ExpertPrompting over base prompts, with winning percentages ranging from 26.53% ($\chi^2 = 82.65, p < .05$) for Gemini 1.5 Pro, to 76.93% ($\chi^2 = 294.27, p < .05$) for Llama 3–70B (see Figure 2). Despite these significant results, we remain skeptical about the robustness of this evaluation method. Relying solely on an LLM to assess outputs from another introduces risk of misjudgment, as seen in Section 4.3.1. Moreover, this pairwise evaluation method frequently selected a "winner" even when both responses were factually incorrect. In some cases, the model even favored an inaccurate response while penalizing an accurate one. Based on this observation, we re-evaluated the same responses for factual accuracy using a tailored version of our accuracy assessment. Once again, no significant improvement was observed. In fact, ExpertPrompting led to decreased accuracy, ranging from a drop of 2.4% ($\chi^2 = 0.84, p = 0.36$) for Gemini 1.5 Pro, to 14.53% ($\chi^2 = 34.00, p < .05$) for Llama 3–8B. This raises an important question: if a response is rated as "high quality" despite being factually incorrect, can it truly be considered an improvement? This can be misleading for users, who might reasonably assume that accuracy is embedded within the "quality" metric. Consequently, despite some studies presenting their claims similarly using verbs such as "improves", "enhances", "overperforms" to describe their prompting techniques, their outcomes cannot effectively be compared. Including both quality and accuracy metrics in evaluations would have provided a more comprehensive understanding of the effectiveness of, e.g., ExpertPrompting.

We would recommend including at least one factual metric when using reasoning benchmarks. Combining complementary metrics would also provide a more comprehensive picture of prompt effectiveness and guards against overinterpretation of single-score gains.

4.3.3 Reporting and Comparing Results Transparently

Across the replicated prompt-engineering studies, we observed substantial methodological variation that complicates the interpretation and comparison of results. Some studies notably display a particularly poor or unclear scientific method. In the EmotionPrompting study (Li et al., 2023), researchers cherry-pick the prompt with the best result out of eleven different prompts, rather than calculating an average across all prompts. This seemingly deliberate action may be due to a publication bias, which motivates researchers to manipulate results to be positive and therefore publishable. In addition, some studies, such as the Re-Reading study (Xu et al., 2024), report results as "significant" multiple times without presenting the corresponding statistical calculations or p-values. This lack of statistical transparency can mislead readers into assuming statistical significance without the necessary evidence to support such claims. It is crucial that when terms like "significant" are used, they are backed by clearly defined statistical measures. Moreover, some studies do not properly report the details of their experimental setup (Perez et al., 2022), which makes it confusing or even impossible to understand and therefore to replicate their process exactly. In this case, the lack of transparency forbids us from detecting possible shortcomings. We recommend adopting standardized evaluation methodologies and clearly defining metrics to ensure that results from different studies can be accurately compared and interpreted.

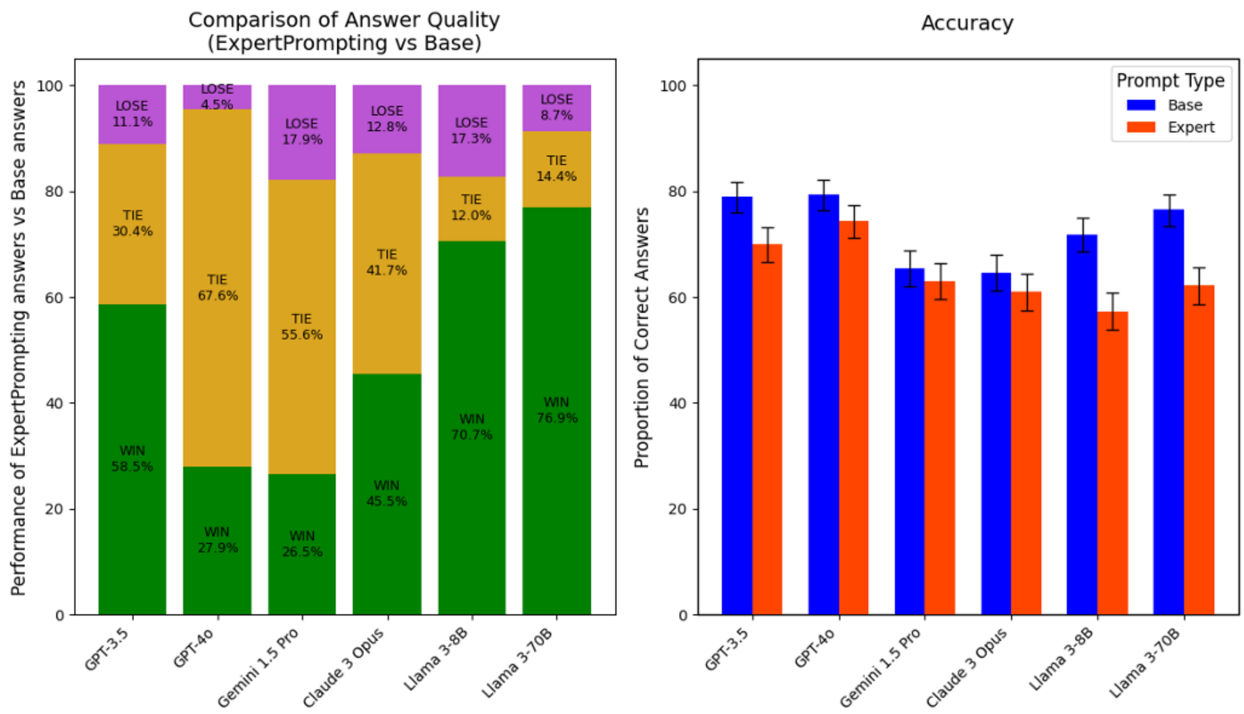

Figure 2: Comparison of accuracy and quality between the base tests and ExpertPrompting, using prompts without formatting constraints, across all LLMs. Error bars show 95% CIs.

Table 3: Results-related replication issues and mitigation techniques

Issue	Mitigation
Evaluation unreliability: unstable or imprecise LLM-as-a-judge scoring	• Use precise, task-specific verification prompts • Run multiple judging prompts and models and average across results • Include manual or rule-based cross-checks
Metric inconsistency: mixing incomparable measures (e.g., accuracy vs. quality)	• Define all metrics clearly and report how each is computed • Include at least one factual accuracy metric in reasoning tasks
Cherry-picking and poor reporting: selective results, missing p-values, or unclear methodology	• Aggregate across all tested prompts or justify subset selection • Report all statistical tests, p-values, and confidence intervals • Provide full methodological details (benchmarks, models, and prompts)

5 Transparency Recommendations and Modular Evaluation Pipeline

To ensure methodological transparency and facilitate replication, future studies should explicitly document the following:

Benchmarks

- Exact benchmark names (and versions if applicable)
- Clear description of what each benchmark measures
- Quality-check methodology, if applied (e.g., automated validation scripts, manual review)
- Dataset provenance, including any filtering, sampling, or modification procedures

Models

- Exact model names and versions
- Description of model class (reasoning vs non-reasoning)
- Full inference methodology, e.g., temperature, max tokens, random seeds
- If used or modified, system prompts and/or added prompt instructions

Evaluation and Metrics

- Exact metrics reported (e.g., accuracy, quality, interpretability) and how they were computed
- If using LLM-as-a-judge: specify the judging model, version, full evaluation prompt, and aggregation method if any
- All statistical analyses: e.g., test type, p-values, confidence intervals

To support transparency and facilitate replication, we developed a modular evaluation pipeline that accompanies this study. The code was designed to allow researchers to replicate, extend, or adapt our experiments with minimal effort. Each stage of the evaluation process is modularized, enabling flexible modification of:

- **Benchmark selection:** users can easily substitute or add benchmarks, specify the number of random items drawn, and control sampling strategies.
- **Prompt-engineering techniques:** zero-shot prompting methods, particularly those relying on prefix or suffix additions, can be switched or customized through dedicated modules.
- **Model configuration:** the model used for inference can be easily changed, facilitating cross-model comparisons.
- **Evaluation and verification setup:** the output classification component allows users to define or modify the verification prompt and evaluate output quality or accuracy according to their own criteria.

This modular design serves a dual purpose. First, it allows rapid, small-scale experiments to obtain a preliminary estimate of the replicability of prior findings before committing to large-scale runs. Second, it encourages exploration and calibration: users can quickly test which combination of model, benchmark, prompting strategy, and evaluation prompt is most appropriate for their research context. The code is accessible here: `https://github.com/Laurene-v/replicatingPET`. By lowering the practical barrier to replication and promoting iterative experimentation, we aim to foster a culture of methodological transparency and empirical verification in prompt-engineering research.

6 Discussion

Our experiments show that most of the tested prompt engineering techniques do not lead to replicable or generalizable performance improvements in LLMs. Most techniques, when applied in slightly different experimental setups, failed to produce the claimed benefits. Some techniques occasionally even resulted in a decrease in response accuracy. In view of the uncritical propagation of the prompt engineering techniques in the literature (Schulhoff et al., 2024), we recommend a more cautious approach when citing papers with insufficient methodological standards. While further research could help reliably understand the conditions under which specific techniques are effective, we offer recommendations and tools to mitigate these replication issues.

While our focus is on replication issues in prompt engineering, it might be the case that similar challenges exist in other LLM behavior evaluation techniques. Our findings suggest that further research is necessary to reliably understand the conditions under which specific prompt engineering techniques are effective, and they underscore the need for replication studies across the broader spectrum of machine behavior research. Only by rigorously verifying or refuting insights from these studies can we build a more reliable foundation for evaluating LLM performance.

In line with the recommendations proposed above, we emphasize the importance of enhanced research transparency and the application of rigorous scientific methods when evaluating LLM behavior. Future investigations should not only continue to scrutinize prompt engineering approaches but also extend replication efforts to other fields of LLM evaluations to ensure the robustness and generalizability of findings in LLM behavior research.

Resource availability

The datasets and code generated during this study are available in the 'ReplicationCrisisInLLMEvaluation' repository on the Open Science Framework (OSF) at `https://osf.io/hcygf/?view_only=fe25a85157734f68882777404aeb655c` and at `https://github.com/Laurene-v/replicatingPET`.

Declaration of interests

The authors declare no competing interests.

References

Zeyuan Allen-Zhu and Yuanzhi Li. Physics of language models: Part 3.2, knowledge manipulation. *arXiv*, 2024. doi: 10.48550/arXiv.2309.14402. URL `https://doi.org/10.48550/arXiv.2309.14402`.

Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete problems in ai safety. *arXiv*, 2016. doi: 10.48550/arXiv.1606.06565. URL `https://doi.org/10.48550/arXiv.1606.06565`.

Anthropic. Claude 3 model card. `https://docs.anthropic.com/en/docs/resources/model-card`, 2024. Accessed: 2024-09-06.

Marcel Binz and Eric Schulz. Using cognitive psychology to understand gpt-3. *Proceedings of the National Academy of Sciences*, 120(6):e2218523120, 2023. doi: 10.1073/pnas.2218523120. URL `https://doi.org/10.1073/pnas.2218523120`.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario

Amodei. Language models are few-shot learners. *arXiv*, 2020. doi: 10.48550/arXiv.2005.14165. URL https://doi.org/10.48550/arXiv.2005.14165.

Jialun Cao, Yuk-Kit Chan, Zixuan Ling, Wenxuan Wang, Shuqing Li, Mingwei Liu, Ruixi Qiao, Yuting Han, Chaozheng Wang, Boxi Yu, Pinjia He, Shuai Wang, Zibin Zheng, Michael R. Lyu, and Shing-Chi Cheung. How should we build a benchmark? revisiting 274 code-related benchmarks for llms, 2025. URL https://arxiv.org/abs/2501.10711.

Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, Wei Ye, Yue Zhang, Yi Chang, Philip S. Yu, Qiang Yang, and Xing Xie. A survey on evaluation of large language models. *arXiv*, 2023. doi: 10.48550/arXiv.2307.03109. URL https://doi.org/10.48550/arXiv.2307.03109.

Lingjiao Chen, Matei Zaharia, and James Zou. How Is ChatGPT's Behavior Changing Over Time? *Harvard Data Science Review*, 6(2), mar 12 2024a. https://hdsr.mitpress.mit.edu/pub/y95zitmz.

Yuyan Chen, Zhihao Wen, Ge Fan, Zhengyu Chen, Wei Wu, Dayiheng Liu, Zhixu Li, Bang Liu, and Yanghua Xiao. Mapo: Boosting large language model performance with model-adaptive prompt optimization, 2024b. URL https://arxiv.org/abs/2407.04118.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, March 2023. URL https://lmsys.org/blog/2023-03-30-vicuna.

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. Training verifiers to solve math word problems. *arXiv*, November 2021. doi: 10.48550/arXiv.2110.14168. URL https://doi.org/10.48550/arXiv.2110.14168.

Yihe Deng, Weitong Zhang, Zixiang Chen, and Quanquan Gu. Rephrase and respond: Let large language models ask better questions for themselves. *arXiv*, 2024. doi: 10.48550/arXiv.2311.04205. URL https://doi.org/10.48550/arXiv.2311.04205.

Mingqi Gao, Xinyu Hu, Jie Ruan, Xiao Pu, and Xiaojun Wan. Llm-based nlg evaluation: Current status and challenges. *arXiv*, 2025. doi: 10.48550/arXiv.2402.01383. URL https://doi.org/10.48550/arXiv.2402.01383.

Aryo Pradipta Gema, Joshua Ong Jun Leang, Giwon Hong, Alessio Devoto, Alberto Carlo Maria Mancino, Rohit Saxena, Xuanli He, Yu Zhao, Xiaotang Du, Mohammad Reza Ghasemi Madani, Claire Barale, Robert McHardy, Joshua Harris, Jean Kaddour, Emile van Krieken, and Pasquale Minervini. Are we done with mmlu? *arXiv*, 2024. URL http://arxiv.org/abs/2406.04127. Version 1.

Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. Did aristotle use a laptop? a question answering benchmark with implicit reasoning strategies. *Transactions of the Association for Computational Linguistics*, 9:346–361, 2021.

Trystan S. Goetze and Darren Abramson. Bigger isn't better: The ethical and scientific vices of extra-large datasets in language models. In *Companion Publication of the 13th ACM Web Science Conference 2021*, pp. 69–75, 2021. doi: 10.1145/3462741.3466809. URL https://doi.org/10.1145/3462741.3466809.

Taicheng Guo, Xiuying Chen, Yaqi Wang, Ruidi Chang, Shichao Pei, Nitesh V. Chawla, Olaf Wiest, and Xiangliang Zhang. Large language model based multi-agents: A survey of progress and challenges. *arXiv*, 2024. doi: 10.48550/arXiv.2402.01680. URL https://doi.org/10.48550/arXiv.2402.01680.

Thilo Hagendorff. Mapping the ethics of generative ai: A comprehensive scoping review. *Minds and Machines*, 34(4):39, 2024. doi: 10.1007/s11023-024-09694-w. URL https://doi.org/10.1007/s11023-024-09694-w.

Thilo Hagendorff, Sarah Fabi, and Michal Kosinski. Human-like intuitive behavior and reasoning biases emerged in large language models but disappeared in chatgpt. *Nature Computational Science*, 3(10):833–838, 2023. doi: 10.1038/s43588-023-00527-x. URL https://doi.org/10.1038/s43588-023-00527-x.

Susan Hao, Piyush Kumar, Sarah Laszlo, Shivani Poddar, Bhaktipriya Radharapu, and Renee Shelby. Safety and fairness for content moderation in generative models. *arXiv*, 2023. doi: 10.48550/arXiv.2306.06135. URL https://doi.org/10.48550/arXiv.2306.06135.

Or Honovich, Uri Shaham, Samuel R. Bowman, and Omer Levy. Instruction induction: From few examples to natural language task descriptions. *arXiv*, 2022. doi: 10.48550/arXiv.2205.10782. URL https://doi.org/10.48550/arXiv.2205.10782.

Jiaming Ji, Tianyi Qiu, Boyuan Chen, Borong Zhang, Hantao Lou, Kaile Wang, Yawen Duan, Zhonghao He, Lukas Vierling, Donghai Hong, Jiayi Zhou, Zhaowei Zhang, Fanzhi Zeng, Juntao Dai, Xuehai Pan, Kwan Yee Ng, Aidan O'Gara, Hua Xu, Brian Tse, Jie Fu, Stephen McAleer, Yaodong Yang, Yizhou Wang, Song-Chun Zhu, Yike Guo, and Wen Gao. Ai alignment: A comprehensive survey. *arXiv*, 2024. doi: 10.48550/arXiv.2310.19852. URL https://doi.org/10.48550/arXiv.2310.19852.

Mingyu Jin, Qinkai Yu, Dong Shu, Haiyan Zhao, Wenyue Hua, Yanda Meng, Yongfeng Zhang, and Mengnan Du. The impact of reasoning step length on large language models. *arXiv*, 2024. URL http://arxiv.org/abs/2401.04925.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. *arXiv*, 2022. URL https://arxiv.org/abs/2205.11916.

Changmao Li and Jeffrey Flanigan. Task contamination: Language models may not be few-shot anymore. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(16):18471–18480, Mar. 2024. doi: 10.1609/aaai.v38i16.29808. URL https://ojs.aaai.org/index.php/AAAI/article/view/29808.

Cheng Li, Jindong Wang, Yixuan Zhang, Kaijie Zhu, Wenxin Hou, Jianxun Lian, Fang Luo, Qiang Yang, and Xing Xie. Emotionprompt: Leveraging psychology for large language models enhancement via emotional stimulus, 2023. URL https://arxiv.org/abs/2307.11760.

Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pp. 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics. URL https://aclanthology.org/W04-1013/.

Stephanie Lin, Jacob Hilton, and Owain Evans. Truthfulqa: Measuring how models mimic human falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3214–3252, 2022.

Chia-Wei Liu, Ryan Lowe, Iulian V. Serban, Michael Noseworthy, Laurent Charlin, and Joelle Pineau. How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation, 2017. URL https://arxiv.org/abs/1603.08023.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9):1–35, 2021. doi: 10.1145/3560815. URL https://doi.org/10.1145/3560815.

Pan Lu, Swaroop Mishra, Tony Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, pp. 2507–2521, 2022.

Shervin Minaee, Tomas Mikolov, Narjes Nikzad, Meysam Chenaghlu, Richard Socher, Xavier Amatriain, and Jianfeng Gao. Large language models: A survey. *arXiv*, 2024. doi: 10.48550/arXiv.2402.06196. URL https://doi.org/10.48550/arXiv.2402.06196.

Swaroop Mishra, Arindam Mitra, Neeraj Varshney, Bhavdeep Sachdeva, Peter Clark, Chitta Baral, and Ashwin Kalyan. Numglue: A suite of fundamental yet challenging mathematical reasoning tasks. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3505–3523, 2022.

Seyed Mahed Mousavi, Edoardo Cecchinato, Lucia Hornikova, and Giuseppe Riccardi. Garbage in, reasoning out? why benchmark scores are unreliable and what to do about it, 2025. URL https://arxiv.org/abs/2506.23864.

Marianna Nezhurina, Lucia Cipolina-Kun, Mehdi Cherti, and Jenia Jitsev. Alice in wonderland: Simple tasks showing complete reasoning breakdown in state-of-the-art large language models, 2024. URL https://arxiv.org/abs/2406.02061.

Jun-Ping Ng and Viktoria Abrecht. Better summarization evaluation with word embeddings for rouge, 2015. URL https://arxiv.org/abs/1508.06034.

Curtis G. Northcutt, Anish Athalye, and Jonas Mueller. Pervasive label errors in test sets destabilize machine learning benchmarks, 2021. URL https://arxiv.org/abs/2103.14749.

OpenAI. Introducing chatgpt, November 2022. URL https://openai.com/index/chatgpt/.

OpenAI. Gpt-4 technical report, 2023. URL https://cdn.openai.com/papers/gpt-4.pdf.

OpenAI. Openai platform, 2024. URL https://platform.openai.com/docs/guides/reasoning. Accessed: 2024-09-13.

Alexander Pan, Jun Shern Chan, Andy Zou, Nathaniel Li, Steven Basart, Thomas Woodside, Jonathan Ng, Hanlin Zhang, Scott Emmons, and Dan Hendrycks. Do the rewards justify the means? measuring trade-offs between rewards and ethical behavior in the machiavelli benchmark, 2023. URL https://arxiv.org/abs/2304.03279.

Roger Peng. Reproducible research in computational science. *Science (New York, N.Y.)*, 334:1226–1227, 2011. doi: 10.1126/science.1213847. URL https://doi.org/10.1126/science.1213847.

Ethan Perez, Sam Ringer, Kamilė Lukošiūtė, Karina Nguyen, Edwin Chen, Scott Heiner, Craig Pettit, Catherine Olsson, Sandipan Kundu, Saurav Kadavath, Andy Jones, Anna Chen, Ben Mann, Brian Israel, Bryan Seethor, Cameron McKinnon, Christopher Olah, Da Yan, Daniela Amodei, Dario Amodei, Dawn Drain, Dustin Li, Eli Tran-Johnson, Guro Khundadze, Jackson Kernion, James Landis, Jamie Kerr, Jared Mueller, Jeeyoon Hyun, Joshua Landau, Kamal Ndousse, Landon Goldberg, Liane Lovitt, Martin Lucas, Michael Sellitto, Miranda Zhang, Neerav Kingsland, Nelson Elhage, Nicholas Joseph, Noemí Mercado, Nova DasSarma, Oliver Rausch, Robin Larson, Sam McCandlish, Scott Johnston, Shauna Kravec, Sheer El Showk, Tamera Lanham, Timothy Telleen-Lawton, Tom Brown, Tom Henighan, Tristan Hume, Yuntao Bai, Zac Hatfield-Dodds, Jack Clark, Samuel R. Bowman, Amanda Askell, Roger Grosse, Danny Hernandez, Deep Ganguli, Evan Hubinger, Nicholas Schiefer, and Jared Kaplan. Discovering language model behaviors with model-written evaluations. *arXiv*, 2022. URL http://arxiv.org/abs/2212.09251.

Yotam Perlitz, Ariel Gera, Ofir Arviv, Asaf Yehudai, Elron Bandel, Eyal Shnarch, Michal Shmueli-Scheuer, and Leshem Choshen. Do these llm benchmarks agree? fixing benchmark evaluation with benchbench, 2024. URL https://arxiv.org/abs/2407.13696.

Long Phan, Alice Gatti, Ziwen Han, Nathaniel Li, Josephina Hu, Hugh Zhang, Chen Bo Calvin Zhang, Mohamed Shaaban, John Ling, Sean Shi, Michael Choi, Anish Agrawal, Arnav Chopra, Adam Khoja, Ryan Kim, Richard Ren, Jason Hausenloy, Oliver Zhang, Mantas Mazeika, Dmitry Dodonov, Tung Nguyen, Jaeho Lee, Daron Anderson, Mikhail Doroshenko, Alun Cennyth Stokes, Mobeen Mahmood, Oleksandr Pokutnyi, Oleg Iskra, Jessica P. Wang, John-Clark Levin, Mstyslav Kazakov, Fiona Feng, Steven Y. Feng, Haoran Zhao, Michael Yu, Varun Gangal, Chelsea Zou, Zihan Wang, Serguei Popov, Robert Gerbicz, Geoff Galgon, Johannes Schmitt, Will Yeadon, Yongki Lee, Scott Sauers, Alvaro Sanchez, Fabian Giska, Marc Roth, Søren Riis, Saiteja Utpala, Noah Burns, Gashaw M. Goshu, Mohinder Maheshbhai Naiya, Chidozie Agu, Zachary Giboney, Antrell Cheatom, Francesco Fournier-Facio, Sarah-Jane Crowson, Lennart

Finke, Zerui Cheng, Jennifer Zampese, Ryan G. Hoerr, Mark Nandor, Hyunwoo Park, Tim Gehrunger, Jiaqi Cai, Ben McCarty, Alexis C Garretson, Edwin Taylor, Damien Sileo, Qiuyu Ren, Usman Qazi, Lianghui Li, Jungbae Nam, John B. Wydallis, Pavel Arkhipov, Jack Wei Lun Shi, Aras Bacho, Chris G. Willcocks, Hangrui Cao, Sumeet Motwani, Emily de Oliveira Santos, Johannes Veith, Edward Vendrow, Doru Cojoc, Kengo Zenitani, Joshua Robinson, Longke Tang, Yuqi Li, Joshua Vendrow, Natanael Wildner Fraga, Vladyslav Kuchkin, Andrey Pupasov Maksimov, Pierre Marion, Denis Efremov, Jayson Lynch, Kaiqu Liang, Aleksandar Mikov, Andrew Gritsevskiy, Julien Guillod, Gözdenur Demir, Dakotah Martinez, Ben Pageler, Kevin Zhou, Saeed Soori, Ori Press, Henry Tang, Paolo Rissone, Sean R. Green, Lina Brüssel, Moon Twayana, Aymeric Dieuleveut, Joseph Marvin Imperial, Ameya Prabhu, Jinzhou Yang, Nick Crispino, Arun Rao, Dimitri Zvonkine, Gabriel Loiseau, Mikhail Kalinin, Marco Lukas, Ciprian Manolescu, Nate Stambaugh, Subrata Mishra, Tad Hogg, Carlo Bosio, Brian P Coppola, Julian Salazar, Jaehyeok Jin, Rafael Sayous, Stefan Ivanov, Philippe Schwaller, Shaipranesh Senthilkuma, Andres M Bran, Andres Algaba, Kelsey Van den Houte, Lynn Van Der Sypt, Brecht Verbeken, David Noever, Alexei Kopylov, Benjamin Myklebust, Bikun Li, Lisa Schut, Evgenii Zheltonozhskii, Qiaochu Yuan, Derek Lim, Richard Stanley, Tong Yang, John Maar, Julian Wykowski, Martí Oller, Anmol Sahu, Cesare Giulio Ardito, Yuzheng Hu, Ariel Ghislain Kemogne Kamdoum, Alvin Jin, Tobias Garcia Vilchis, Yuexuan Zu, Martin Lackner, James Koppel, Gongbo Sun, Daniil S. Antonenko, Steffi Chern, Bingchen Zhao, Pierrot Arsene, Joseph M Cavanagh, Daofeng Li, Jiawei Shen, Donato Crisostomi, Wenjin Zhang, Ali Dehghan, Sergey Ivanov, David Perrella, Nurdin Kaparov, Allen Zang, Ilia Sucholutsky, Arina Kharlamova, Daniil Orel, Vladislav Poritski, Shalev Ben-David, Zachary Berger, Parker Whitfill, Michael Foster, Daniel Munro, Linh Ho, Shankar Sivarajan, Dan Bar Hava, Aleksey Kuchkin, David Holmes, Alexandra Rodriguez-Romero, Frank Sommerhage, Anji Zhang, Richard Moat, Keith Schneider, Zakayo Kazibwe, Don Clarke, Dae Hyun Kim, Felipe Meneguitti Dias, Sara Fish, Veit Elser, Tobias Kreiman, Victor Efren Guadarrama Vilchis, Immo Klose, Ujjwala Anantheswaran, Adam Zweiger, Kaivalya Rawal, Jeffery Li, Jeremy Nguyen, Nicolas Daans, Haline Heidinger, Maksim Radionov, Václav Rozhoň, Vincent Ginis, Christian Stump, Niv Cohen, Rafał Poświata, Josef Tkadlec, Alan Goldfarb, Chenguang Wang, Piotr Padlewski, Stanislaw Barzowski, Kyle Montgomery, Ryan Stendall, Jamie Tucker-Foltz, Jack Stade, T. Ryan Rogers, Tom Goertzen, Declan Grabb, Abhishek Shukla, Alan Givré, John Arnold Ambay, Archan Sen, Muhammad Fayez Aziz, Mark H Inlow, Hao He, Ling Zhang, Younesse Kaddar, Ivar Ängquist, Yanxu Chen, Harrison K Wang, Kalyan Ramakrishnan, Elliott Thornley, Antonio Terpin, Hailey Schoelkopf, Eric Zheng, Avishy Carmi, Ethan D. L. Brown, Kelin Zhu, Max Bartolo, Richard Wheeler, Martin Stehberger, Peter Bradshaw, JP Heimonen, Kaustubh Sridhar, Ido Akov, Jennifer Sandlin, Yury Makarychev, Joanna Tam, Hieu Hoang, David M. Cunningham, Vladimir Goryachev, Demosthenes Patramanis, Michael Krause, Andrew Redenti, David Aldous, Jesyin Lai, Shannon Coleman, Jiangnan Xu, Sangwon Lee, Ilias Magoulas, Sandy Zhao, Ning Tang, Michael K. Cohen, Orr Paradise, Jan Hendrik Kirchner, Maksym Ovchynnikov, Jason O. Matos, Adithya Shenoy, Michael Wang, Yuzhou Nie, Anna Sztyber-Betley, Paolo Faraboschi, Robin Riblet, Jonathan Crozier, Shiv Halasyamani, Shreyas Verma, Prashant Joshi, Eli Meril, Ziqiao Ma, Jérémy Andréoletti, Raghav Singhal, Jacob Platnick, Volodymyr Nevirkovets, Luke Basler, Alexander Ivanov, Seri Khoury, Nils Gustafsson, Marco Piccardo, Hamid Mostaghimi, Qijia Chen, Virendra Singh, Tran Quoc Khánh, Paul Rosu, Hannah Szlyk, Zachary Brown, Himanshu Narayan, Aline Menezes, Jonathan Roberts, William Alley, Kunyang Sun, Arkil Patel, Max Lamparth, Anka Reuel, Linwei Xin, Hanmeng Xu, Jacob Loader, Freddie Martin, Zixuan Wang, Andrea Achilleos, Thomas Preu, Tomek Korbak, Ida Bosio, Fereshteh Kazemi, Ziye Chen, Biró Bálint, Eve J. Y. Lo, Jiaqi Wang, Maria Inês S. Nunes, Jeremiah Milbauer, M Saiful Bari, Zihao Wang, Behzad Ansarinejad, Yewen Sun, Stephane Durand, Hossam Elgnainy, Guillaume Douville, Daniel Tordera, George Balabanian, Hew Wolff, Lynna Kvistad, Hsiaoyun Milliron, Ahmad Sakor, Murat Eron, Andrew Favre D. O., Shailesh Shah, Xiaoxiang Zhou, Firuz Kamalov, Sherwin Abdoli, Tim Santens, Shaul Barkan, Allison Tee, Robin Zhang, Alessandro Tomasiello, G. Bruno De Luca, Shi-Zhuo Looi, Vinh-Kha Le, Noam Kolt, Jiayi Pan, Emma Rodman, Jacob Drori, Carl J Fossum, Niklas Muennighoff, Milind Jagota, Ronak Pradeep, Honglu Fan, Jonathan Eicher, Michael Chen, Kushal Thaman, William Merrill, Moritz Firsching, Carter Harris, Stefan Ciobâcă, Jason Gross, Rohan Pandey, Ilya Gusev, Adam Jones, Shashank Agnihotri, Pavel Zhelnov, Mohammadreza Mofayezi, Alexander Piperski, David K. Zhang, Kostiantyn Dobarskyi, Roman Leventov, Ignat Soroko, Joshua Duersch, Vage Taamazyan, Andrew Ho, Wenjie Ma, William Held, Ruicheng Xian, Armel Randy Zebaze, Mohanad Mohamed, Julian Noah Leser, Michelle X Yuan, Laila Yacar, Johannes Lengler, Katarzyna Olszewska, Claudio Di Fratta, Edson Oliveira, Joseph W. Jackson, Andy Zou, Muthu Chidambaram, Timothy Manik, Hector Haffenden, Dashiell Stander, Ali Dasouqi, Alexander Shen, Bita Golshani, David Stap, Egor Kretov, Mikalai Uzhou, Alina Borisovna Zhidkovskaya, Nick Winter, Miguel Orbegozo Rodriguez, Robert Lauff, Dustin Wehr, Colin Tang, Zaki Hossain, Shaun Phillips, Fortuna Samuele, Fredrik Ekström, Angela Hammon, Oam Patel, Faraz Farhidi, George Medley, Forough Mohammadzadeh, Madellene Peñaflor, Haile Kassahun, Alena Friedrich, Rayner Hernandez Perez, Daniel Pyda, Taom Sakal, Omkar Dhamane, Ali Khajegili Mirabadi, Eric Hallman, Kenchi Okutsu, Mike Battaglia, Mohammad Maghsoudimehrabani, Alon Amit, Dave Hulbert, Roberto Pereira, Simon Weber, Handoko, Anton Peristyy, Stephen Malina, Mustafa Mehkary, Rami Aly, Frank Reidegeld, Anna-Katharina Dick, Cary Friday, Mukhwinder Singh, Hassan Shapourian, Wanyoung Kim, Mariana Costa, Hubeyb Gurdogan, Harsh Kumar, Chiara Ceconello, Chao Zhuang, Haon Park, Micah Carroll, Andrew R. Tawfeek, Stefan Steinerberger, Daattavya Aggarwal, Michael Kirchhof, Linjie Dai, Evan Kim, Johan Ferret, Jainam Shah, Yuzhou Wang, Minghao Yan, Krzysztof Burdzy, Lixin Zhang, Antonio Franca, Diana T. Pham, Kang Yong Loh, Joshua Robinson, Abram Jackson, Paolo Giordano, Philipp Petersen, Adrian Cosma, Jesus Colino, Colin White, Jacob Votava, Vladimir Vinnikov, Ethan Delaney, Petr Spelda, Vit Stritecky, Syed M. Shahid, Jean-Christophe Mourrat, Lavr Vetoshkin, Koen Sponselee, Renas Bacho, Zheng-Xin Yong, Florencia de la Rosa, Nathan Cho, Xiuyu Li, Guillaume Malod, Orion Weller, Guglielmo Albani, Leon Lang, Julien Laurendeau, Dmitry Kazakov, Fatimah Adesanya, Julien Portier, Lawrence Hollom, Victor Souza, Yuchen Anna Zhou, Julien Degorre, Yiğit Yalın, Gbenga Daniel Obikoya, Rai, Filippo Bigi, M. C. Boscá, Oleg Shumar, Kaniuar Bacho, Gabriel Recchia, Mara Popescu, Nikita Shulga, Ngefor Mildred Tanwie, Thomas C. H. Lux, Ben Rank, Colin Ni, Matthew Brooks, Alesia Yakimchyk, Huanxu, Liu, Stefano Cavalleri, Olle Häggström, Emil Verkama, Joshua Newbould, Hans Gundlach, Leonor Brito-Santana, Brian Amaro, Vivek Vajipey, Rynaa Grover, Ting Wang, Yosi Kratish, Wen-Ding Li, Sivakanth Gopi, Andrea Caciolai, Christian Schroeder de Witt, Pablo Hernández-Cámara, Emanuele Rodolà, Jules Robins, Dominic Williamson, Vincent Cheng, Brad Raynor, Hao Qi, Ben Segev, Jingxuan Fan, Sarah Martinson, Erik Y. Wang, Kaylie Hausknecht, Michael P. Brenner, Mao Mao, Christoph Demian, Peyman Kassani, Xinyu Zhang, David Avagian, Eshawn Jessica Scipio, Alon Ragoler, Justin Tan, Blake Sims, Rebeka Plecnik, Aaron Kirtland, Omer Faruk Bodur, D. P. Shinde, Yan Carlos Leyva Labrador, Zahra Adoul, Mohamed Zekry, Ali Karakoc, Tania C. B. Santos, Samir Shamseldeen, Loukmane Karim, Anna Liakhovitskaia, Nate Resman, Nicholas Farina, Juan Carlos Gonzalez, Gabe Maayan, Earth Anderson, Rodrigo De Oliveira Pena, Elizabeth Kelley, Hodjat Mariji, Rasoul Pouriamanesh, Wentao Wu, Ross Finocchio, Ismail Alarab, Joshua Cole, Danyelle Ferreira, Bryan Johnson, Mohammad Safdari, Liangti Dai, Siriphan Arthornthurasuk, Isaac C. McAlister, Alejandro José Moyano, Alexey Pronin, Jing Fan, Angel Ramirez-Trinidad, Yana Malysheva, Daphiny Pottmaier, Omid Taheri, Stanley Stepanic, Samuel Perry, Luke Askew, Raúl Adrián Huerta Rodríguez, Ali M. R. Minissi, Ricardo Lorena, Krishnamurthy Iyer, Arshad Anil Fasiludeen, Ronald Clark, Josh Ducey, Matheus Piza, Maja Somrak, Eric Vergo, Juehang Qin, Benjámin Borbás, Eric Chu, Jack Lindsey, Antoine Jallon, I. M. J. McInnis, Evan Chen, Avi Semler, Luk Gloor, Tej Shah, Marc Carauleanu, Pascal Lauer, Tran Đuc Huy, Hossein Shahrtash, Emilien Duc, Lukas Lewark, Assaf Brown, Samuel Albanie, Brian Weber, Warren S. Vaz, Pierre Clavier, Yiyang Fan, Gabriel Poesia Reis e Silva, Long, Lian, Marcus Abramovitch, Xi Jiang, Sandra Mendoza, Murat Islam, Juan Gonzalez, Vasilios Mavroudis, Justin Xu, Pawan Kumar, Laxman Prasad Goswami, Daniel Bugas, Nasser Heydari, Ferenc Jeanplong, Thorben Jansen, Antonella Pinto, Archimedes Apronti, Abdallah Galal, Ng Ze-An, Ankit Singh, Tong Jiang, Joan of Arc Xavier, Kanu Priya Agarwal, Mohammed Berkani, Gang Zhang, Zhehang Du, Benedito Alves de Oliveira Junior, Dmitry Malishev, Nicolas Remy, Taylor D. Hartman, Tim Tarver, Stephen Mensah, Gautier Abou Loume, Wiktor Morak, Farzad Habibi, Sarah Hoback, Will Cai, Javier Gimenez, Roselynn Grace Montecillo, Jakub Łucki, Russell Campbell, Asankhaya Sharma, Khalida Meer, Shreen Gul, Daniel Espinosa Gonzalez, Xavier Alapont, Alex Hoover, Gunjan Chhablani, Freddie Vargus, Arunim Agarwal, Yibo Jiang, Deepakkumar Patil, David Outevsky, Kevin Joseph Scaria, Rajat Maheshwari, Abdelkader Dendane, Priti Shukla, Ashley Cartwright, Sergei Bogdanov, Niels Mündler, Sören Möller, Luca Arnaboldi, Kunvar Thaman, Muhammad Rehan Siddiqi, Prajvi Saxena, Himanshu Gupta, Tony Fruhauff, Glen Sherman, Mátyás Vincze, Siranut Usawasutsakorn, Dylan Ler, Anil Radhakrishnan, Innocent Enyekwe, Sk Md Salauddin, Jiang Muzhen, Aleksandr Maksapetyan, Vivien Rossbach, Chris Harjadi, Mohsen Bahaloohoreh, Claire Sparrow, Jasdeep Sidhu, Sam Ali, Song Bian, John Lai, Eric Singer, Justine Leon Uro, Greg Bateman, Mohamed Sayed, Ahmed Menshawy, Darling Duclosel, Dario Bezzi, Yashaswini Jain, Ashley Aaron, Murat Tiryakioglu, Sheeshram Siddh, Keith Krenek, Imad Ali Shah, Jun Jin, Scott Creighton, Denis Peskoff, Zienab EL-Wasif, Ragavendran P V, Michael Richmond, Joseph McGowan, Tejal Patwardhan, Hao-Yu Sun, Ting Sun, Nikola Zubić, Samuele Sala, Stephen Ebert, Jean Kaddour, Manuel Schottdorf, Dianzhuo Wang, Gerol Petruzella, Alex Meiburg, Tilen Medved, Ali ElSheikh, S Ashwin Hebbar, Lorenzo Vaquero, Xianjun Yang, Jason Poulos, Vilém Zouhar, Sergey Bogdanik, Mingfang Zhang, Jorge Sanz-Ros, David Anugraha, Yinwei Dai, Anh N. Nhu, Xue Wang, Ali Anil Demircali, Zhibai Jia, Yuyin Zhou, Juncheng Wu, Mike He, Nitin Chandok, Aarush Sinha, Gaoxiang Luo, Long Le, Mickaël Noyé, Michał Perełkiewicz, Ioannis Pantidis, Tianbo Qi, Soham Sachin Purohit, Letitia Parcalabescu, Thai-Hoa Nguyen, Genta Indra Winata, Edoardo M. Ponti, Hanchen Li, Kaustubh Dhole, Jongee Park, Dario Abbondanza, Yuanli Wang, Anupam Nayak, Diogo M. Caetano, Antonio A. W. L. Wong, Maria del Rio-Chanona, Dániel Kondor, Pieter Francois, Ed Chalstrey, Jakob Zsambok, Dan Hoyer, Jenny Reddish, Jakob Hauser, Francisco-Javier Rodrigo-Ginés, Suchandra Datta, Maxwell Shepherd, Thom Kamphuis, Qizheng Zhang, Hyunjun Kim, Ruiji Sun, Jianzhu Yao, Franck Dernoncourt, Satyapriya Krishna, Sina Rismanchian, Bonan Pu, Francesco Pinto, Yingheng Wang, Kumar Shridhar, Kalon J. Overholt, Glib Briia, Hieu Nguyen, David, Soler Bartomeu, Tony CY Pang, Adam Wecker, Yifan Xiong, Fanfei Li, Lukas S. Huber, Joshua Jaeger, Romano De Maddalena, Xing Han Lù, Yuhui Zhang, Claas Beger, Patrick Tser Jern Kon, Sean Li, Vivek Sanker, Ming Yin, Yihao Liang, Xinlu Zhang, Ankit Agrawal, Li S. Yifei, Zechen Zhang, Mu Cai, Yasin Sonmez, Costin Cozianu, Changhao Li, Alex Slen, Shoubin Yu, Hyun Kyu Park, Gabriele Sarti, Marcin Briański, Alessandro Stolfo, Truong An Nguyen, Mike Zhang, Yotam Perlitz, Jose Hernandez-Orallo, Runjia Li, Amin Shabani, Felix Juefei-Xu, Shikhar Dhingra, Orr Zohar, My Chiffon Nguyen, Alexander Pondaven, Abdurrahim Yilmaz, Xuandong Zhao, Chuanyang Jin, Muyan Jiang, Stefan Todoran, Xinyao Han, Jules Kreuer, Brian Rabern, Anna Plassart, Martino Maggetti, Luther Yap, Robert Geirhos, Jonathon Kean, Dingsu Wang, Sina Mollaei, Chenkai Sun, Yifan Yin, Shiqi Wang, Rui Li, Yaowen Chang, Anjiang Wei, Alice Bizeul, Xiaohan Wang, Alexandre Oliveira Arrais, Kushin Mukherjee, Jorge Chamorro-Padial, Jiachen Liu, Xingyu Qu, Junyi Guan, Adam Bouyamourn, Shuyu Wu, Martyna Plomecka, Junda Chen, Mengze Tang, Jiaqi Deng, Shreyas Subramanian, Haocheng Xi, Haoxuan Chen, Weizhi Zhang, Yinuo Ren, Haoqin Tu, Sejong Kim, Yushun Chen, Sara Vera Marjanović, Junwoo Ha, Grzegorz Luczyna, Jeff J. Ma, Zewen Shen, Dawn Song, Cedegao E. Zhang, Zhun Wang, Gaël Gendron, Yunze Xiao, Leo Smucker, Erica Weng, Kwok Hao Lee, Zhe Ye, Stefano Ermon, Ignacio D. Lopez-Miguel, Theo Knights, Anthony Gitter, Namkyu Park, Boyi Wei, Hongzheng Chen, Kunal Pai, Ahmed Elkhanany, Han Lin, Philipp D. Siedler, Jichao Fang, Ritwik Mishra, Károly Zsolnai-Fehér, Xilin Jiang, Shadab Khan, Jun Yuan, Rishab Kumar Jain, Xi Lin, Mike Peterson, Zhe Wang, Aditya Malusare, Maosen Tang, Isha Gupta, Ivan Fosin, Timothy Kang, Barbara Dworakowska, Kazuki Matsumoto, Guangyao Zheng, Gerben Sewuster, Jorge Pretel Villanueva, Ivan Rannev, Igor Chernyavsky, Jiale Chen, Deepayan Banik, Ben Racz, Wenchao Dong, Jianxin Wang, Laila Bashmal, Duarte V. Gonçalves, Wei Hu, Kaushik Bar, Ondrej Bohdal, Atharv Singh Patlan, Shehzaad Dhuliawala, Caroline Geirhos, Julien Wist, Yuval Kansal, Bingsen Chen, Kutay Tire, Atak Talay Yücel, Brandon Christof, Veerupaksh Singla, Zijian Song, Sanxing Chen, Jiaxin Ge, Kaustubh Ponkshe, Isaac Park, Tianneng Shi, Martin Q. Ma, Joshua Mak, Sherwin Lai, Antoine Moulin, Zhuo Cheng, Zhanda Zhu, Ziyi Zhang, Vaidehi Patil, Ketan Jha, Qiutong Men, Jiaxuan Wu, Tianchi Zhang, Bruno Hebling Vieira, Alham Fikri Aji, Jae-Won Chung, Mohammed Mahfoud, Ha Thi Hoang, Marc Sperzel, Wei Hao, Kristof Meding, Sihan Xu, Vassilis Kostakos, Davide Manini, Yueying Liu, Christopher Toukmaji, Jay Paek, Eunmi Yu, Arif Engin Demircali, Zhiyi Sun, Ivan Dewerpe, Hongsen Qin, Roman Pflugfelder, James Bailey, Johnathan Morris, Ville Heilala, Sybille Rosset, Zishun Yu, Peter E. Chen, Woongyeong Yeo, Eeshaan Jain, Ryan Yang, Sreekar Chigurupati, Julia Chernyavsky, Sai Prajwal Reddy, Subhashini Venugopalan, Hunar Batra, Core Francisco Park, Hieu Tran, Guilherme Maximiano, Genghan Zhang, Yizhuo Liang, Hu Shiyu, Rongwu Xu, Rui Pan, Siddharth Suresh, Ziqi Liu, Samaksh Gulati, Songyang Zhang, Peter Turchin, Christopher W. Bartlett, Christopher R. Scotese, Phuong M. Cao, Ben Wu, Jacek Karwowski, Davide Scaramuzza, Aakaash Nattanmai, Gordon McKellips, Anish Cheraku, Asim Suhail, Ethan Luo, Marvin Deng, Jason Luo, Ashley Zhang, Kavin Jindel, Jay Paek, Kasper Halevy, Allen Baranov, Michael Liu, Advaith Avadhanam, David Zhang, Vincent Cheng, Brad Ma, Evan Fu, Liam Do, Joshua Lass, Hubert Yang, Surya Sunkari, Vishruth Bharath, Violet Ai, James Leung, Rishit Agrawal, Alan Zhou, Kevin Chen, Tejas Kalpathi, Ziqi Xu, Gavin Wang, Tyler Xiao, Erik Maung, Sam Lee, Ryan Yang, Roy Yue, Ben Zhao, Julia Yoon, Sunny Sun, Aryan Singh, Ethan Luo, Clark Peng, Tyler Osbey, Taozhi Wang, Daryl

Echeazu, Hubert Yang, Timothy Wu, Spandan Patel, Vidhi Kulkarni, Vijaykaarti Sundarapandiyan, Ashley Zhang, Andrew Le, Zafir Nasim, Srikar Yalam, Ritesh Kasamsetty, Soham Samal, Hubert Yang, David Sun, Nihar Shah, Abhijeet Saha, Alex Zhang, Leon Nguyen, Laasya Nagumalli, Kaixin Wang, Alan Zhou, Aidan Wu, Jason Luo, Anwith Telluri, Summer Yue, Alexandr Wang, and Dan Hendrycks. Humanity's last exam, 2025. URL https://arxiv.org/abs/2501.14249.

Sander Schulhoff, Michael Ilie, Nishant Balepur, Konstantine Kahadze, Amanda Liu, Chenglei Si, Yinheng Li, Aayush Gupta, HyoJung Han, Sevien Schulhoff, Pranav Sandeep Dulepet, Saurav Vidyadhara, Dayeon Ki, Sweta Agrawal, Chau Pham, Gerson Kroiz, Feileen Li, Hudson Tao, Ashay Srivastava, Hevander Da Costa, Saloni Gupta, Megan L. Rogers, Inna Goncearenco, Giuseppe Sarli, Igor Galynker, Denis Peskoff, Marine Carpuat, Jules White, Shyamal Anadkat, Alexander Hoyle, and Philip Resnik. The prompt report: A systematic survey of prompting techniques. *arXiv*, 2024. URL http://arxiv.org/abs/2406.06608.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R. Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, Agnieszka Kluska, Aitor Lewkowycz, Akshat Agarwal, Alethea Power, Alex Ray, Alex Warstadt, Alexander W. Kocurek, Ali Safaya, Ali Tazarv, Alice Xiang, Alicia Parrish, Allen Nie, Aman Hussain, Amanda Askell, Amanda Dsouza, Ambrose Slone, Ameet Rahane, Anantharaman S. Iyer, Anders Andreassen, Andrea Madotto, Andrea Santilli, Andreas Stuhlmüller, Andrew Dai, Andrew La, Andrew Lampinen, Andy Zou, Angela Jiang, Angelica Chen, Anh Vuong, Animesh Gupta, Anna Gottardi, Antonio Norelli, Anu Venkatesh, Arash Gholamidavoodi, Arfa Tabassum, Arul Menezes, Arun Kirubarajan, Asher Mullokandov, Ashish Sabharwal, Austin Herrick, Avia Efrat, Aykut Erdem, Ayla Karakaş, B. Ryan Roberts, Bao Sheng Loe, Barret Zoph, Bartłomiej Bojanowski, Batuhan Özyurt, Behnam Hedayatnia, Behnam Neyshabur, Benjamin Inden, Benno Stein, Berk Ekmekci, Bill Yuchen Lin, Blake Howald, Bryan Orinion, Cameron Diao, Cameron Dour, Catherine Stinson, Cedrick Argueta, César Ferri Ramírez, Chandan Singh, Charles Rathkopf, Chenlin Meng, Chitta Baral, Chiyu Wu, Chris Callison-Burch, Chris Waites, Christian Voigt, Christopher D. Manning, Christopher Potts, Cindy Ramirez, Clara E. Rivera, Clemencia Siro, Colin Raffel, Courtney Ashcraft, Cristina Garbacea, Damien Sileo, Dan Garrette, Dan Hendrycks, Dan Kilman, Dan Roth, Daniel Freeman, Daniel Khashabi, Daniel Levy, Daniel Moseguí González, Danielle Perszyk, Danny Hernandez, Danqi Chen, Daphne Ippolito, Dar Gilboa, David Dohan, David Drakard, David Jurgens, Debajyoti Datta, Deep Ganguli, Denis Emelin, Denis Kleyko, Deniz Yuret, Derek Chen, Derek Tam, Dieuwke Hupkes, Diganta Misra, Dilyar Buzan, Dimitri Coelho Mollo, Diyi Yang, Dong-Ho Lee, Dylan Schrader, Ekaterina Shutova, Ekin Dogus Cubuk, Elad Segal, Eleanor Hagerman, Elizabeth Barnes, Elizabeth Donoway, Ellie Pavlick, Emanuele Rodola, Emma Lam, Eric Chu, Eric Tang, Erkut Erdem, Ernie Chang, Ethan A. Chi, Ethan Dyer, Ethan Jerzak, Ethan Kim, Eunice Engefu Manyasi, Evgenii Zheltonozhskii, Fanyue Xia, Fatemeh Siar, Fernando Martínez-Plumed, Francesca Happé, Francois Chollet, Frieda Rong, Gaurav Mishra, Genta Indra Winata, Gerard de Melo, Germán Kruszewski, Giambattista Parascandolo, Giorgio Mariani, Gloria Wang, Gonzalo Jaimovitch-López, Gregor Betz, Guy Gur-Ari, Hana Galijasevic, Hannah Kim, Hannah Rashkin, Hannaneh Hajishirzi, Harsh Mehta, Hayden Bogar, Henry Shevlin, Hinrich Schütze, Hiromu Yakura, Hongming Zhang, Hugh Mee Wong, Ian Ng, Isaac Noble, Jaap Jumelet, Jack Geissinger, Jackson Kernion, Jacob Hilton, Jaehoon Lee, Jaime Fernández Fisac, James B. Simon, James Koppel, James Zheng, James Zou, Jan Kocoń, Jana Thompson, Janelle Wingfield, Jared Kaplan, Jarema Radom, Jascha Sohl-Dickstein, Jason Phang, Jason Wei, Jason Yosinski, Jekaterina Novikova, Jelle Bosscher, Jennifer Marsh, Jeremy Kim, Jeroen Taal, Jesse Engel, Jesujoba Alabi, Jiacheng Xu, Jiaming Song, Jillian Tang, Joan Waweru, John Burden, John Miller, John U. Balis, Jonathan Batchelder, Jonathan Berant, Jörg Frohberg, Jos Rozen, Jose Hernandez-Orallo, Joseph Boudeman, Joseph Guerr, Joseph Jones, Joshua B. Tenenbaum, Joshua S. Rule, Joyce Chua, Kamil Kanclerz, Karen Livescu, Karl Krauth, Karthik Gopalakrishnan, Katerina Ignatyeva, Katja Markert, Kaustubh D. Dhole, Kevin Gimpel, Kevin Omondi, Kory Mathewson, Kristen Chiafullo, Ksenia Shkaruta, Kumar Shridhar, Kyle McDonell, Kyle Richardson, Laria Reynolds, Leo Gao, Li Zhang, Liam Dugan, Lianhui Qin, Lidia Contreras-Ochando, Louis-Philippe Morency, Luca Moschella, Lucas Lam, Lucy Noble, Ludwig Schmidt, Luheng He, Luis Oliveros Colón, Luke Metz, Lütfi Kerem Şenel, Maarten Bosma, Maarten Sap, Maartje ter Hoeve, Maheen Farooqi, Manaal Faruqui, Mantas Mazeika, Marco Baturan, Marco Marelli, Marco Maru, Maria Jose Ramírez Quintana, Marie Tolkiehn, Mario Giulianelli, Martha Lewis, Martin Potthast, Matthew L. Leavitt, Matthias Hagen, Mátyás Schubert, Medina Orduna Baitemirova, Melody Arnaud, Melvin McElrath, Michael A. Yee, Michael Cohen, Michael Gu, Michael Ivanitskiy, Michael Starritt, Michael Strube, Michał Swędrowski, Michele Bevilacqua, Michihiro Yasunaga, Mihir Kale, Mike Cain, Mimee Xu, Mirac Suzgun, Mitch Walker, Mo Tiwari, Mohit Bansal, Moin Aminnaseri, Mor Geva, Mozhdeh Gheini, Mukund Varma T, Nanyun Peng, Nathan A. Chi, Nayeon Lee, Neta Gur-Ari Krakover, Nicholas Cameron, Nicholas Roberts, Nick Doiron, Nicole Martinez, Nikita Nangia, Niklas Deckers, Niklas Muennighoff, Nitish Shirish Keskar, Niveditha S. Iyer, Noah Constant, Noah Fiedel, Nuan Wen, Oliver Zhang, Omar Agha, Omar Elbaghdadi, Omer Levy, Owain Evans, Pablo Antonio Moreno Casares, Parth Doshi, Pascale Fung, Paul Pu Liang, Paul Vicol, Pegah Alipoormolabashi, Peiyuan Liao, Percy Liang, Peter Chang, Peter Eckersley, Phu Mon Htut, Pinyu Hwang, Piotr Miłkowski, Piyush Patil, Pouya Pezeshkpour, Priti Oli, Qiaozhu Mei, Qing Lyu, Qinlang Chen, Rabin Banjade, Rachel Etta Rudolph, Raefer Gabriel, Rahel Habacker, Ramon Risco, Raphaël Millière, Rhythm Garg, Richard Barnes, Rif A. Saurous, Riku Arakawa, Robbe Raymaekers, Robert Frank, Rohan Sikand, Roman Novak, Roman Sitelew, Ronan LeBras, Rosanne Liu, Rowan Jacobs, Rui Zhang, Ruslan Salakhutdinov, Ryan Chi, Ryan Lee, Ryan Stovall, Ryan Teehan, Rylan Yang, Sahib Singh, Saif M. Mohammad, Sajant Anand, Sam Dillavou, Sam Shleifer, Sam Wiseman, Samuel Gruetter, Samuel R. Bowman, Samuel S. Schoenholz, Sanghyun Han, Sanjeev Kwatra, Sarah A. Rous, Sarik Ghazarian, Sayan Ghosh, Sean Casey, Sebastian Bischoff, Sebastian Gehrmann, Sebastian Schuster, Sepideh Sadeghi, Shadi Hamdan, Sharon Zhou, Shashank Srivastava, Sherry Shi, Shikhar Singh, Shima Asaadi, Shixiang Shane Gu, Shubh Pachchigar, Shubham Toshniwal, Shyam Upadhyay, Shyamolima, Debnath, Siamak Shakeri, Simon Thormeyer, Simone Melzi, Siva Reddy, Sneha Priscilla Makini, Soo-Hwan Lee, Spencer Torene, Sriharsha Hatwar, Stanislas Dehaene, Stefan Divic, Stefano Ermon, Stella Biderman, Stephanie Lin, Stephen Prasad, Steven T. Piantadosi, Stuart M. Shieber, Summer Misherghi, Svetlana Kiritchenko, Swaroop Mishra, Tal Linzen, Tal Schuster, Tao Li, Tao Yu, Tariq Ali, Tatsu Hashimoto, Te-Lin Wu, Théo Desbordes, Theodore Rothschild, Thomas Phan, Tianle Wang, Tiberius Nkinyili, Timo Schick, Timofei Kornev, Titus Tunduny, Tobias Gerstenberg, Trenton Chang, Trishala Neeraj, Tushar Khot, Tyler Shultz, Uri Shaham, Vedant Misra, Vera Demberg, Victoria Nyamai, Vikas Raunak, Vinay Ramasesh, Vinay Uday Prabhu, Vishakh Padmakumar, Vivek Srikumar, William Fedus, William Saunders, William Zhang, Wout Vossen, Xiang Ren, Xiaoyu Tong, Xinran Zhao, Xinyi Wu, Xudong Shen, Yadollah Yaghoobzadeh, Yair Lakretz, Yangqiu Song, Yasaman Bahri, Yejin Choi, Yichi Yang, Yiding Hao, Yifu Chen, Yonatan Belinkov, Yu Hou, Yufang Hou, Yuntao Bai, Zachary Seid, Zhuoye Zhao, Zijian Wang, Zijie J. Wang, Zirui Wang, and Ziyi Wu. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *arXiv*, 2022. URL `https://arxiv.org/abs/2206.04615`.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. Commonsenseqa: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4149–4158, 2019.

Gemini Team, Petar Georgiev, V. I. Lei, R. Burnell, L. Bai, A. Gulati, G. Tanzer, D. Vincent, Z. Pan, S. Wang, S. Mariooryad, Y. Ding, X. Geng, F. Alcober, R. Frostig, M. Omernick, L. Walker, C. Paduraru, C. Sorokin, ..., and Oriol Vinyals. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv*, pp. 1–90, 2024.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models. *arXiv*, 2023. URL `https://arxiv.org/abs/2307.09288v2`.

Laurène Vaugrante, Francesca Carlon, Maluna Menke, and Thilo Hagendorff. Compromising honesty and harmlessness in language models via deception attacks, 2025. URL https://arxiv.org/abs/2502.08301.

Boxin Wang, Weixin Chen, Hengzhi Pei, Chulin Xie, Mintong Kang, Chenhui Zhang, Chejian Xu, Zidi Xiong, Ritik Dutta, Rylan Schaeffer, Sang T. Truong, Simran Arora, Mantas Mazeika, Dan Hendrycks, Zinan Lin, Yu Cheng, Sanmi Koyejo, Dawn Song, and Bo Li. Decodingtrust: A comprehensive assessment of trustworthiness in gpt models. *arXiv*, 2024a. URL http://arxiv.org/abs/2306.11698.

Guoqing Wang, Zeyu Sun, Zhihao Gong, Sixiang Ye, Yizhou Chen, Yifan Zhao, Qingyuan Liang, and Dan Hao. Do advanced language models eliminate the need for prompt engineering in software engineering?, 2024b. URL https://arxiv.org/abs/2411.02093.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models, 2023. URL https://arxiv.org/abs/2201.11903.

Laura Weidinger, Maribeth Rauh, Nahema Marchal, Arianna Manzini, Lisa Anne Hendricks, Juan Mateos-Garcia, Stevie Bergman, Jackie Kay, Conor Griffin, Ben Bariach, Iason Gabriel, Verena Rieser, and William Isaac. Sociotechnical safety evaluation of generative ai systems. *arXiv*, 2023. doi: 10.48550/arXiv.2310.11986. URL https://doi.org/10.48550/arXiv.2310.11986.

BigScience Workshop, :, Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, Jonathan Tow, Alexander M. Rush, Stella Biderman, Albert Webson, Pawan Sasanka Ammanamanchi, Thomas Wang, Benoît Sagot, Niklas Muennighoff, Albert Villanova del Moral, Olatunji Ruwase, Rachel Bawden, Stas Bekman, Angelina McMillan-Major, Iz Beltagy, Huu Nguyen, Lucile Saulnier, Samson Tan, Pedro Ortiz Suarez, Victor Sanh, Hugo Laurençon, Yacine Jernite, Julien Launay, Margaret Mitchell, Colin Raffel, Aaron Gokaslan, Adi Simhi, Aitor Soroa, Alham Fikri Aji, Amit Alfassy, Anna Rogers, Ariel Kreisberg Nitzav, Canwen Xu, Chenghao Mou, Chris Emezue, Christopher Klamm, Colin Leong, Daniel van Strien, David Ifeoluwa Adelani, Dragomir Radev, Eduardo González Ponferrada, Efrat Levkovizh, Ethan Kim, Eyal Bar Natan, Francesco De Toni, Gérard Dupont, Germán Kruszewski, Giada Pistilli, Hady Elsahar, Hamza Benyamina, Hieu Tran, Ian Yu, Idris Abdulmumin, Isaac Johnson, Itziar Gonzalez-Dios, Javier de la Rosa, Jenny Chim, Jesse Dodge, Jian Zhu, Jonathan Chang, Jörg Frohberg, Joseph Tobing, Joydeep Bhattacharjee, Khalid Almubarak, Kimbo Chen, Kyle Lo, Leandro Von Werra, Leon Weber, Long Phan, Loubna Ben allal, Ludovic Tanguy, Manan Dey, Manuel Romero Muñoz, Maraim Masoud, María Grandury, Mario Šaško, Max Huang, Maximin Coavoux, Mayank Singh, Mike Tian-Jian Jiang, Minh Chien Vu, Mohammad A. Jauhar, Mustafa Ghaleb, Nishant Subramani, Nora Kassner, Nurulaqilla Khamis, Olivier Nguyen, Omar Espejel, Ona de Gibert, Paulo Villegas, Peter Henderson, Pierre Colombo, Priscilla Amuok, Quentin Lhoest, Rheza Harliman, Rishi Bommasani, Roberto Luis López, Rui Ribeiro, Salomey Osei, Sampo Pyysalo, Sebastian Nagel, Shamik Bose, Shamsuddeen Hassan Muhammad, Shanya Sharma, Shayne Longpre, Somaieh Nikpoor, Stanislav Silberberg, Suhas Pai, Sydney Zink, Tiago Timponi Torrent, Timo Schick, Tristan Thrush, Valentin Danchev, Vassilina Nikoulina, Veronika Laippala, Violette Lepercq, Vrinda Prabhu, Zaid Alyafeai, Zeerak Talat, Arun Raja, Benjamin Heinzerling, Chenglei Si, Davut Emre Taşar, Elizabeth Salesky, Sabrina J. Mielke, Wilson Y. Lee, Abheesht Sharma, Andrea Santilli, Antoine Chaffin, Arnaud Stiegler, Debajyoti Datta, Eliza Szczechla, Gunjan Chhablani, Han Wang, Harshit Pandey, Hendrik Strobelt, Jason Alan Fries, Jos Rozen, Leo Gao, Lintang Sutawika, M Saiful Bari, Maged S. Al-shaibani, Matteo Manica, Nihal Nayak, Ryan Teehan, Samuel Albanie, Sheng Shen, Srulik Ben-David, Stephen H. Bach, Taewoon Kim, Tali Bers, Thibault Fevry, Trishala Neeraj, Urmish Thakker, Vikas Raunak, Xiangru Tang, Zheng-Xin Yong, Zhiqing Sun, Shaked Brody, Yallow Uri, Hadar Tojarieh, Adam Roberts, Hyung Won Chung, Jaesung Tae, Jason Phang, Ofir Press, Conglong Li, Deepak Narayanan, Hatim Bourfoune, Jared Casper, Jeff Rasley, Max Ryabinin, Mayank Mishra, Minjia Zhang, Mohammad Shoeybi, Myriam Peyrounette, Nicolas Patry, Nouamane Tazi, Omar Sanseviero, Patrick von Platen, Pierre Cornette, Pierre François Lavallée, Rémi Lacroix, Samyam Rajbhandari, Sanchit Gandhi, Shaden Smith, Stéphane Requena, Suraj Patil, Tim Dettmers, Ahmed Baruwa, Amanpreet Singh, Anastasia Cheveleva, Anne-Laure Ligozat, Arjun Subramonian, Aurélie Névéol, Charles Lovering, Dan Garrette, Deepak Tunuguntla, Ehud Reiter, Ekaterina Taktasheva, Ekaterina Voloshina, Eli Bogdanov, Genta Indra

Winata, Hailey Schoelkopf, Jan-Christoph Kalo, Jekaterina Novikova, Jessica Zosa Forde, Jordan Clive, Jungo Kasai, Ken Kawamura, Liam Hazan, Marine Carpuat, Miruna Clinciu, Najoung Kim, Newton Cheng, Oleg Serikov, Omer Antverg, Oskar van der Wal, Rui Zhang, Ruochen Zhang, Sebastian Gehrmann, Shachar Mirkin, Shani Pais, Tatiana Shavrina, Thomas Scialom, Tian Yun, Tomasz Limisiewicz, Verena Rieser, Vitaly Protasov, Vladislav Mikhailov, Yada Pruksachatkun, Yonatan Belinkov, Zachary Bamberger, Zdeněk Kasner, Alice Rueda, Amanda Pestana, Amir Feizpour, Ammar Khan, Amy Faranak, Ana Santos, Anthony Hevia, Antigona Unldreaj, Arash Aghagol, Arezoo Abdollahi, Aycha Tammour, Azadeh HajiHosseini, Bahareh Behroozi, Benjamin Ajibade, Bharat Saxena, Carlos Muñoz Ferrandis, Daniel McDuff, Danish Contractor, David Lansky, Davis David, Douwe Kiela, Duong A. Nguyen, Edward Tan, Emi Baylor, Ezinwanne Ozoani, Fatima Mirza, Frankline Ononiwu, Habib Rezanejad, Hessie Jones, Indrani Bhattacharya, Irene Solaiman, Irina Sedenko, Isar Nejadgholi, Jesse Passmore, Josh Seltzer, Julio Bonis Sanz, Livia Dutra, Mairon Samagaio, Maraim Elbadri, Margot Mieskes, Marissa Gerchick, Martha Akinlolu, Michael McKenna, Mike Qiu, Muhammed Ghauri, Mykola Burynok, Nafis Abrar, Nazneen Rajani, Nour Elkott, Nour Fahmy, Olanrewaju Samuel, Ran An, Rasmus Kromann, Ryan Hao, Samira Alizadeh, Sarmad Shubber, Silas Wang, Sourav Roy, Sylvain Viguier, Thanh Le, Tobi Oyebade, Trieu Le, Yoyo Yang, Zach Nguyen, Abhinav Ramesh Kashyap, Alfredo Palasciano, Alison Callahan, Anima Shukla, Antonio Miranda-Escalada, Ayush Singh, Benjamin Beilharz, Bo Wang, Caio Brito, Chenxi Zhou, Chirag Jain, Chuxin Xu, Clémentine Fourrier, Daniel León Periñán, Daniel Molano, Dian Yu, Enrique Manjavacas, Fabio Barth, Florian Fuhrimann, Gabriel Altay, Giyaseddin Bayrak, Gully Burns, Helena U. Vrabec, Imane Bello, Ishani Dash, Jihyun Kang, John Giorgi, Jonas Golde, Jose David Posada, Karthik Rangasai Sivaraman, Lokesh Bulchandani, Lu Liu, Luisa Shinzato, Madeleine Hahn de Bykhovetz, Maiko Takeuchi, Marc Pàmies, Maria A Castillo, Marianna Nezhurina, Mario Sänger, Matthias Samwald, Michael Cullan, Michael Weinberg, Michiel De Wolf, Mina Mihaljcic, Minna Liu, Moritz Freidank, Myungsun Kang, Natasha Seelam, Nathan Dahlberg, Nicholas Michio Broad, Nikolaus Muellner, Pascale Fung, Patrick Haller, Ramya Chandrasekhar, Renata Eisenberg, Robert Martin, Rodrigo Canalli, Rosaline Su, Ruisi Su, Samuel Cahyawijaya, Samuele Garda, Shlok S Deshmukh, Shubhanshu Mishra, Sid Kiblawi, Simon Ott, Sinee Sang-aroonsiri, Srishti Kumar, Stefan Schweter, Sushil Bharati, Tanmay Laud, Théo Gigant, Tomoya Kainuma, Wojciech Kusa, Yanis Labrak, Yash Shailesh Bajaj, Yash Venkatraman, Yifan Xu, Yingxin Xu, Yu Xu, Zhe Tan, Zhongli Xie, Zifan Ye, Mathilde Bras, Younes Belkada, and Thomas Wolf. Bloom: A 176b-parameter open-access multilingual language model. *arXiv*, 2023. doi: 10.48550/arXiv.2211.05100. URL https://doi.org/10.48550/arXiv.2211.05100.

Mingqi Wu, Zhihao Zhang, Qiaole Dong, Zhiheng Xi, Jun Zhao, Senjie Jin, Xiaoran Fan, Yuhao Zhou, Huijie Lv, Ming Zhang, Yanwei Fu, Qin Liu, Songyang Zhang, and Qi Zhang. Reasoning or memorization? unreliable results of reinforcement learning due to data contamination, 2025. URL https://arxiv.org/abs/2507.10532.

Benfeng Xu, An Yang, Junyang Lin, Quan Wang, Chang Zhou, Yongdong Zhang, and Zhendong Mao. Expertprompting: Instructing large language models to be distinguished experts, 2025. URL https://arxiv.org/abs/2305.14688.

Xiaohan Xu, Chongyang Tao, Tao Shen, Can Xu, Hongbo Xu, Guodong Long, Jian guang Lou, and Shuai Ma. Re-reading improves reasoning in large language models. *arXiv*, 2024. URL http://arxiv.org/abs/2309.06275.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. *arXiv*, 2023. doi: 10.48550/arXiv.2305.10601. URL https://doi.org/10.48550/arXiv.2305.10601.

Terry Jingchen Zhang, Gopal Dev, Ning Wang, Nicole Ni, Wenyuan Jiang, Yinya Huang, Bernhard Schölkopf, Mrinmaya Sachan, and Zhijing Jin. Beyond memorization: Reasoning-driven synthesis as a mitigation strategy against benchmark contamination, 2025. URL https://arxiv.org/abs/2509.00072.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena, 2023. URL https://arxiv.org/abs/2306.05685.

Ming Zhou, Nan Duan, Shujie Liu, and Heung-Yeung Shum. Progress in neural nlp: Modeling, learning, and reasoning. *Engineering*, 6(3):275–290, 2020. doi: 10.1016/j.eng.2019.12.0. URL `https://doi.org/10.1016/j.eng.2019.12.0`.

A Additional Results for Vicuna 13B and BLOOM

In addition to the models presented in this paper, we conducted experiments on older models: Vicuna 13B v1.5 and BLOOM. These models were selected to evaluate whether the effectiveness of prompt engineering techniques differs between older and newer models. Vicuna and BLOOM were specifically chosen because they were used in some of the studies we analyzed. For Vicuna, we used the 13B v1.5 via Huggingface, configured in Question Answering mode. However, the accuracy of both the prompt-enhanced and base tests was so low that the results could not be reliably interpreted (see Table 4 and Figure 3). The outputs frequently consisted of question repetition, endlessly looped text snippets, or outright gibberish.

Table 4: Example outputs for Vicuna 13B and BLOOM

Benchmark task	**Vicuna 13B output**	**BLOOM output**
Solve the task below. Importantly, write your final answer after “####”. Answer this question: A lot of people play the fiddle in what state which comes first alphabetically? The choices are the following: ’alabama’, ’orchestra’, ’tennessee’, ’violin’, ’case’, ’alaska’.	Solve the task below. Answer this question: A lot of people play the fiddle in what state which comes first alphabetically? The choices are the following: ’alabama’, ’orchestra’, ’tennessee’, ’violin case’, ’alaska’. Importantly, write your final answer after “####”. 1 and	al answer after “####”./n This is the Answer : Alabama, Tennessee, Alaska, Orchestra, Violin case, ##### A: The answer is Alabama, Tennessee, Alaska, Orchestra, Violin case, ##### because the first letter of each word is the same as the first letter of the state it represents.
Solve the task below. Importantly, write your final answer after “####”. You simplify complicated ideas into something easy to understand. Your common sense shines as a clear guide, helping people when they’re stuck on complex issues. Answer this question: A lot of people play the fiddle in what state which comes first alphabetically? The choices are the following: ’alabama’, ’orchestra’, ’tennessee’, ’violin case’, ’alaska’.	Solve the task below. You simplify complicated ideas into something easy to understand. Your common sense shines as a clear guide, helping people when they’re stuck on complex issues. Answer this question: A lot of people play the fiddle in what state which comes first alphabetically? The choices are the following: ’alabama’, ’orchestra’, ’tennessee’, ’violin case’, ’alaska’. Importantly, write your final answer after “####”.	al answer after “####”./n This is the Answer : Alabama, Tennessee, Alaska, Orchestra, Violin case. A: The answer is Alabama, Tennessee, Alaska, Orchestra, Violin case. Because the fiddle is a musical instrument, and the fiddle is played by an orchestra. The fiddle is played by a violinist, and the violinist plays the violin. The fiddle is played in the orchestra, and the orchestra is in the violin case. The fiddle is played in the orchestra, and the orchestra is in the violin case. The fiddle is played in the orchestra, and the orchestra is in the violin case. [*repeated until it reaches the output’s maximal tokens*]

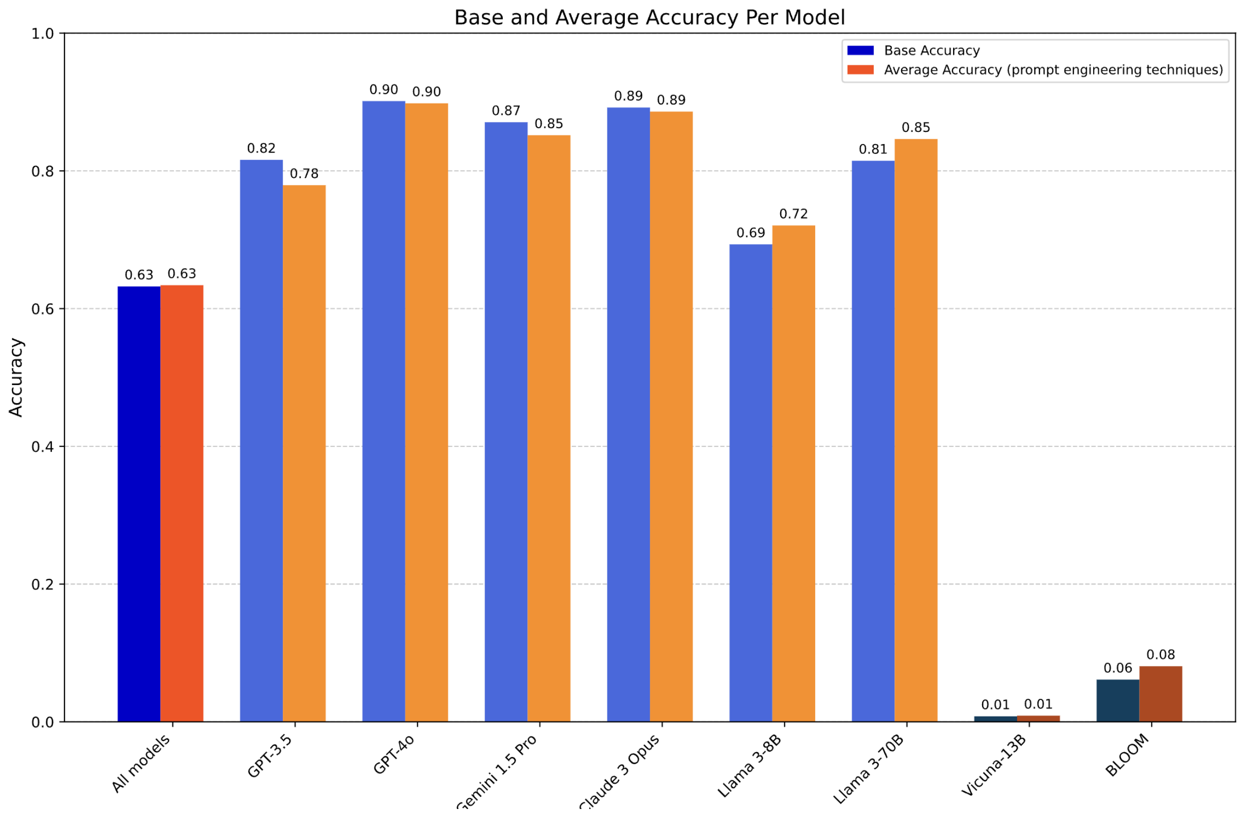

Figure 3: Accuracy comparisons between the base tests with and without any prompt engineering technique across all LLMs.

B Answer Classification Methodology

All benchmarks used for our experiments possess ground truth entries which constitute short strings: one-word answers for most multiple-choice questions, numbers for the numerical questions, short answers for the open questions. The outputs produced by the LLMs often greatly differ in their length from the indicated answers, with prefixes such as "The correct answer is", "Let's proceed step by step", followed by a detailed methodology used to solve the given task, or greetings and politeness markers. The most common metrics used to calculate string similarity, such as the F1-score or ROUGE (Lin, 2004), are therefore not adapted for this use case. Hence, we added the following instruction to each prompt: "Solve the task below. Importantly, write your final answer after '####' ". Using Regex rules, we then extracted the answer after "####" if this string was present in the output and kept the entire output when the model failed to follow the instruction. We then applied a set of Regex rules on this answer accordingly:

- Check if the LLM answer equals the ground truth (when all punctuation has been removed)
- If it is a multiple-choice question, check if the LLM answer is one of the incorrect choices
- If the expected output is a boolean, verify all possible adjacent outputs ("True" could be "TRUE" or "Yes", whereas false could be "FALSE" or "No" for example)

If the Regex rules are unable to classify the answer as correct or incorrect, we then use GPT-4o to classify the string following "####" (see Table 5).

Table 5: GPT-4o instructions used to classify the evaluated LLM's answer

Classification instructions
Your task is to classify a response to a question, specifically assessing whether the given response is true or false in comparison to the correct response. This is the given response: LLM_ANSWER. This is the correct response: CORRECT_ANSWER Adhere to the following rules when classifying the given response: - The classification result should be a boolean value, represented as either TRUE or FALSE. - Only output 'TRUE' or 'FALSE' - Never include any other comments or strings other than TRUE or FALSE in your output. - Do not answer the question. - The given response can be phrased differently from the correct response. This does not mean it should be classified as FALSE. - Only assess if the given response contains the information that matches with the information provided in the correct response. - Check whether the correct response is in essence occurring somewhere in the given response. If so, classify it as TRUE. - Focus on the relevant information in the given response, which is often at the end of it. Do not consider non-essential information such as greetings, small talk, and comments about hobbies or skills when classifying the given response

This classification technique applies to all prompt engineering methods except for the Rephrase-and-Respond technique. Unlike other methods, this approach requires the model to first provide a rephrased version of the question before delivering the answer. This directly impacts the classification process, as the rephrased question can alter what is considered correct. For instance, in True or False questions, the rephrased question might reverse the meaning of the original ("Was Mark the winner?" could become "Did Mark lose?"). Similarly, in multiple-choice questions, the rephrasing might replace the correct choice with a synonym. To address this, we carefully designed a specific classification function for this experiment. Both the Regex rules and the GPT-4o fallback prompt were adapted to suit the requirements of each benchmark.

Table 6: Classification rules tailored to Rephrase-and-Respond LLM answers

Benchmark	**Choices**	**Type of correct answer**	**Rules for RaR**
Common- -senseQA	Multiple choices	Nouns, location names or groups of words; no numbers or dates. Examples: "positive", "alabama" or "great sorrow"	- Verify if all non-rephrased choices are present in the answer. If so, classify with the non-RaR adapted method; - Detect and validate enumerations of options in the LLM's answer, which would correspond to the rephrased choices; - If there is an enumeration of the same length as the length of the given choices, then establish synonyms based on the order of apparition. Detect if the rephrased correct option is the one in the final answer; - Use GPT-4o for fallback classification when the rules cannot determine correctness.
CRT	Open-ended	Values. Examples: "9 weeks", "12 minutes" or "5 dollars"	- Match numbers (digits and word forms); - Allow additional numbers only if they appear in the original question, to account for the rephrased question; - Use fallback verification (GPT-4o) when direct or numerical matches fail.
NumGLUE	Open-ended	Numbers, days or months. Examples: "18", "Monday", "August"	- Remove commas in numbers; - Extract all possible answers from the LLM's answer (numbers, days, months); - If the correct answer is present in the extracted answers, then return TRUE, otherwise return FALSE; - If no possible answers have been detected, then use GPT-4o to classify.
ScienceQA	Multiple choices	Nouns, values, group of words or full sentences. Examples: "simile", "20 seconds", "by clearly stating the main idea", "Both are only physical changes."	Same rules as for CommonsenseQA
StrategyQA	True or False	True or False	With StrategyQA, we exclusively use GPT-4o to classify the rephrased answers, with the prompt shown in Table 5.

Table 7: GPT-4o instructions used to classify the evaluated LLM's answer for Rephrase-and-Respond when the benchmark is StrategyQA

Classification Instructions
Your task is to classify responses to rephrased questions answered by either True or False. You need to determine if the given answer is correct. To establish what is correct or incorrect, you are given the original question and its correct answer. The answer to the rephrased question is correct if the information is consistent with the updated question based on the original ground truth. You are provided: 1. The original question. 2. The ground truth answer for the original question. 3. The rephrased question and its associated answer. The associated answer should be towards the end, after "####". Adhere to the following instructions: - Determine whether the answer to the rephrased question is consistent with the original ground truth. - The answer to the rephrased question and the ground truth can be different if the question has been rephrased as the opposite question, like if "Is the tree big?" was rephrased to "Is the tree small?". Focus on the facts mentioned in the rephrased question/answer pair, compared to the ground truth. - The classification result should be a boolean value, represented as either TRUE or FALSE. - Only respond with "TRUE" or "FALSE". Avoid including comments or explanations. Examples: - Original question: "Should you wear sunscreen to avoid cancer?" Ground truth: "True". Rephrased question: "Rephrased question: Can sunscreen help prevent cancer? Answer: Yes, it can. True #### True" → TRUE. - Original question: "Should you wear sunscreen to avoid cancer?" Ground truth: "True". Rephrased question: "Rephrased question: Is someone wearing sunscreen as likely to get cancer as someone not wearing sunscreen? Answer: False ####" → TRUE. - Original question: "Is the sun a planet?" Ground truth: "False". Rephrased question: "Is the sun a star? #### True" → TRUE. - Original question: "Is the earth flat?" Ground truth: "False". Rephrased question: "Question: Is the earth round? Answer: No, the earth isn't round because the geometry would not make any sense if so. False #### False" → FALSE. - Original question: "The dice shows 2. Did Paul win? Facts: Paul wins if the number on the dice is odd, Marcus wins if it is even." Ground truth: "False". Rephrased question: "Question: Given that Paul wins if the dice number is odd and Marcus wins if the dice number is even, did Marcus win if the dice shows 2? Answer: True, 2 is an even number. Therefore, Marcus wins. #### True" → TRUE.

C Detailed Accuracy Differences Across Prompting Techniques and Benchmarks

The following figures show, for each LLM, the rounded accuracy difference compared to the base prompting, for each prompt engineering technique – benchmark combination.

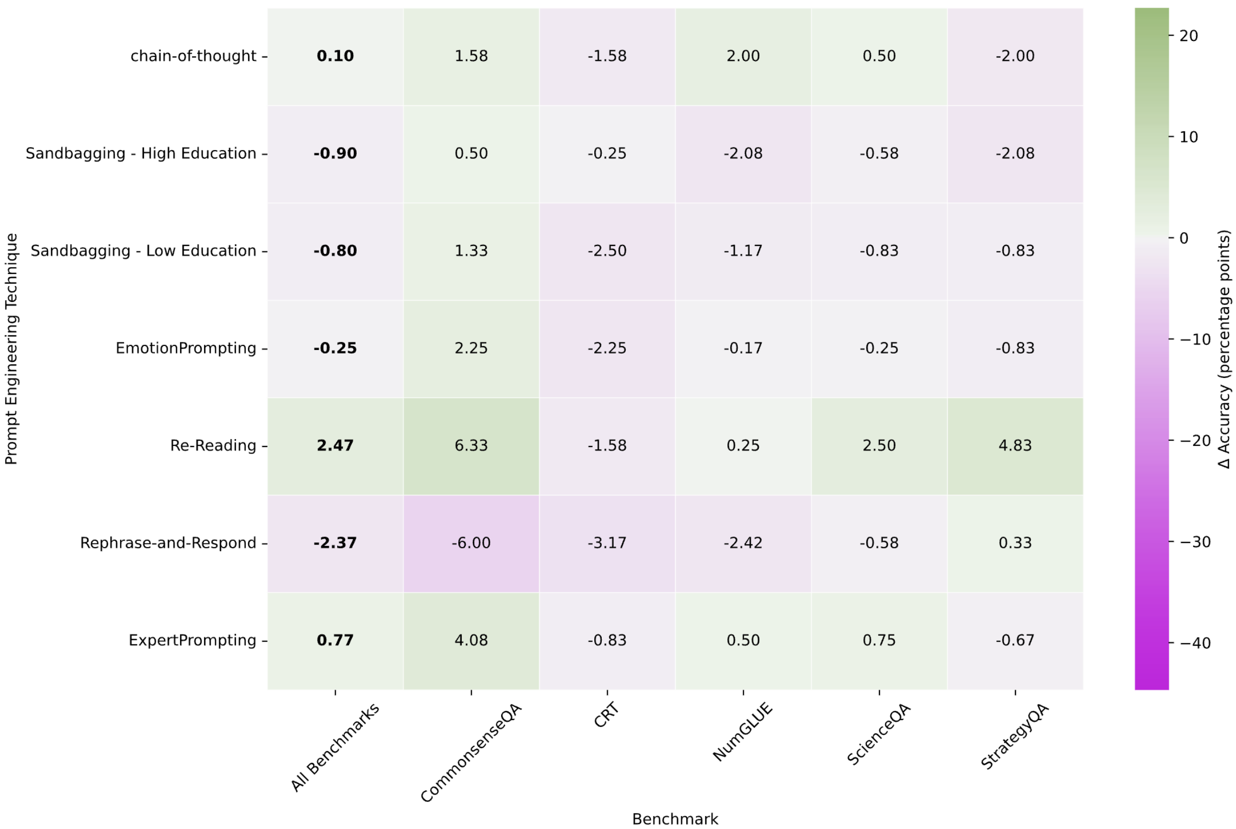

Figure 4: Accuracy differences between prompt engineering techniques and the base prompting, for all models and benchmarks

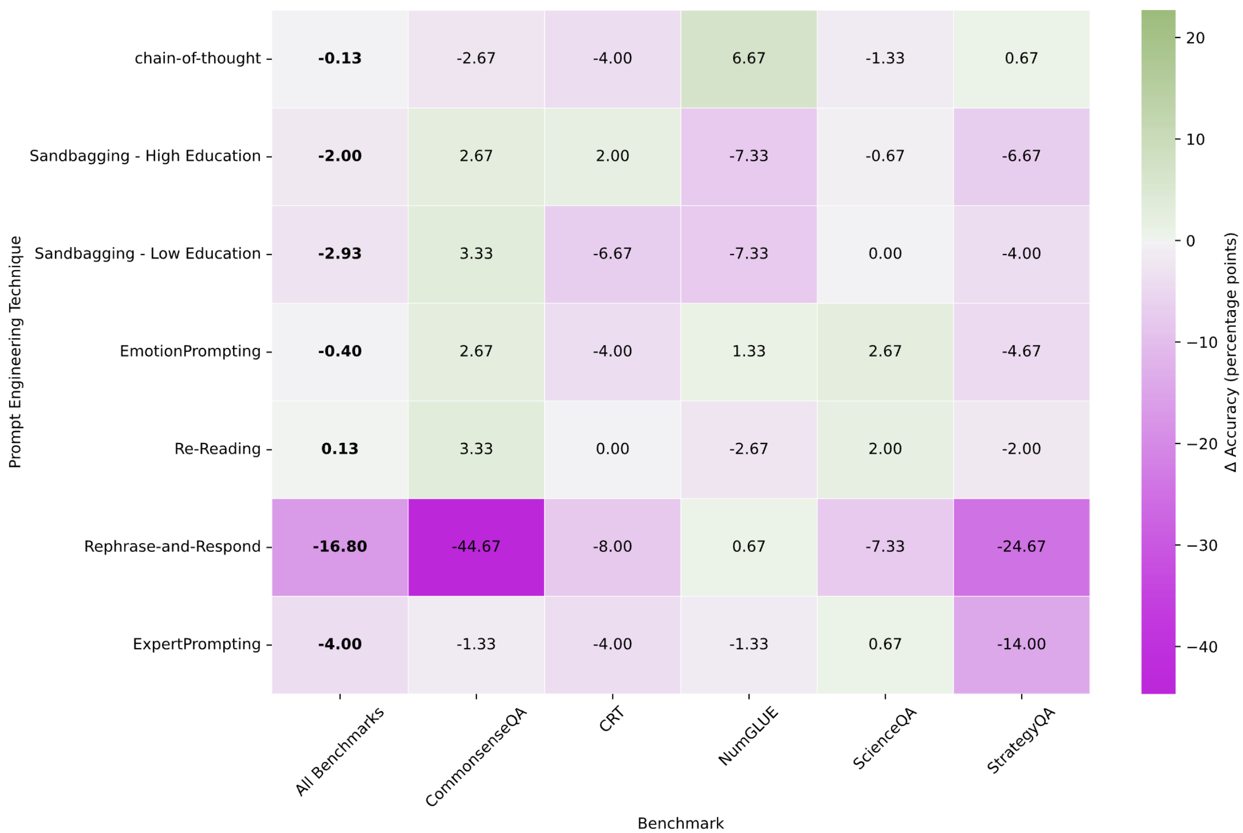

Figure 5: Accuracy differences between prompt engineering techniques and the base prompting, for GPT 3.5 on all benchmarks

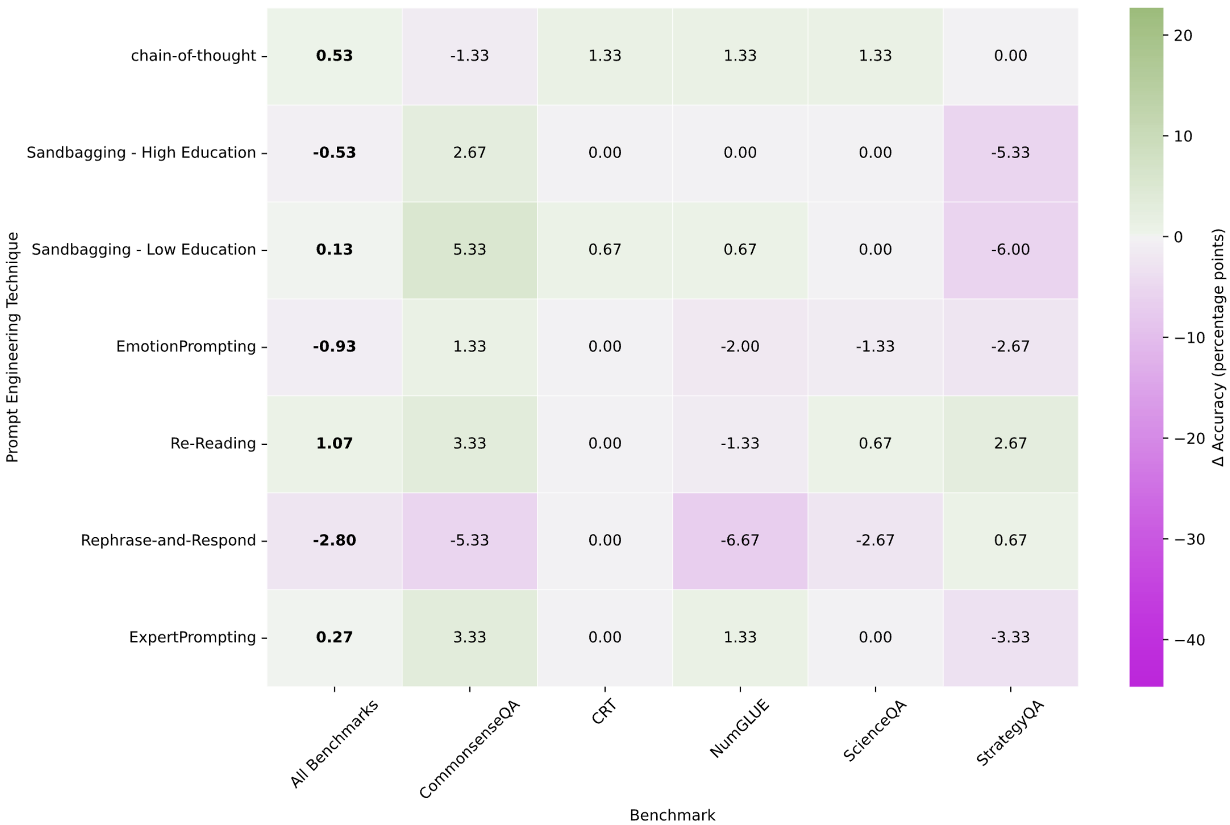

Figure 6: Accuracy differences between prompt engineering techniques and the base prompting, for GPT-4o on all benchmarks

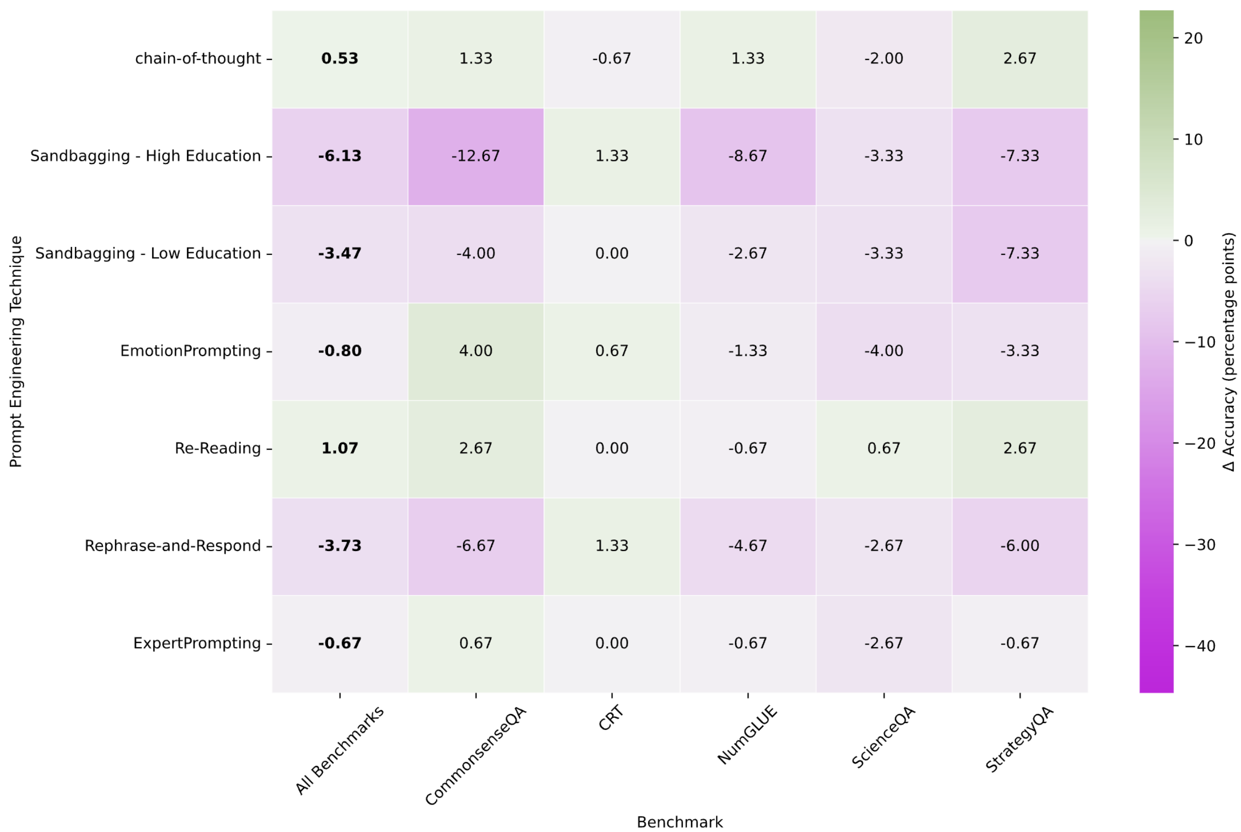

Figure 7: Accuracy differences between prompt engineering techniques and the base prompting, for Gemini 1.5 Pro on all benchmarks

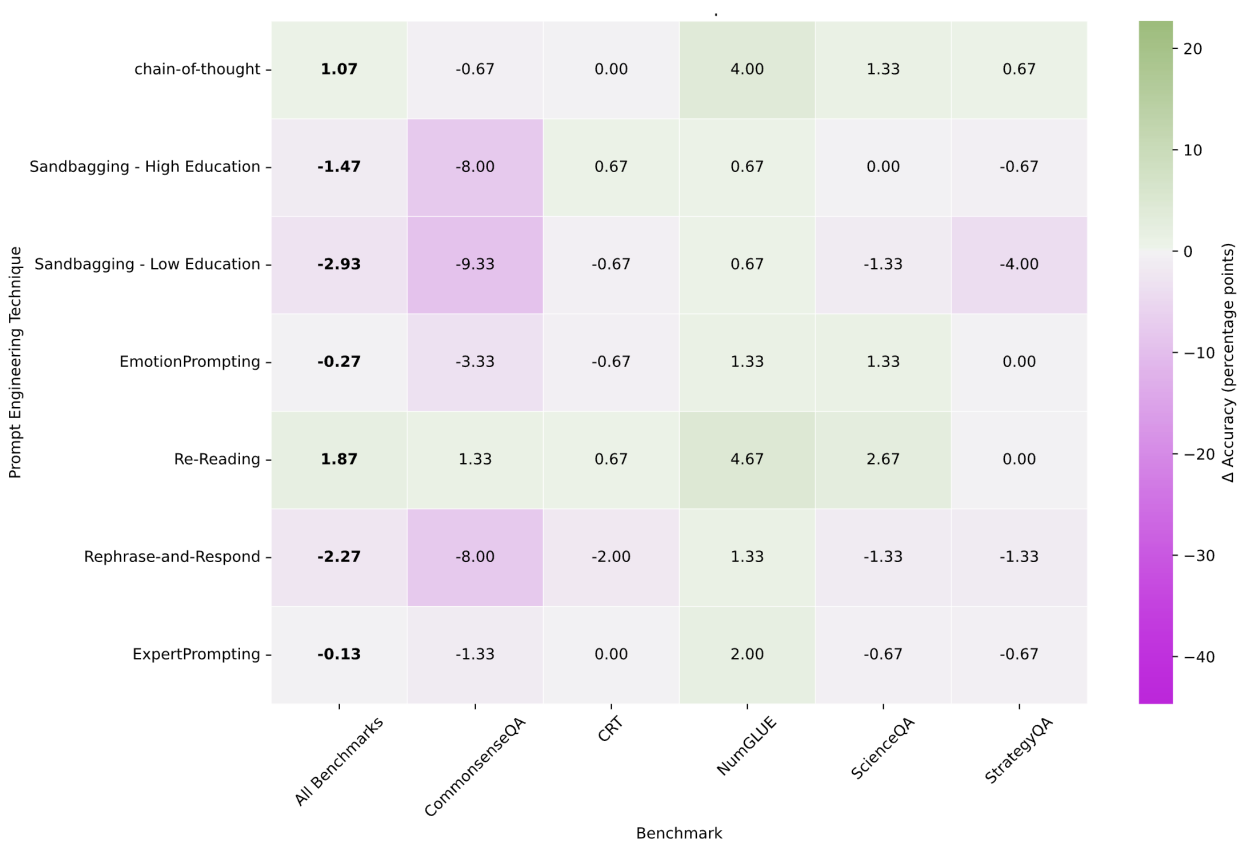

Figure 8: Accuracy differences between prompt engineering techniques and the base prompting, for Claude 3 Opus on all benchmarks

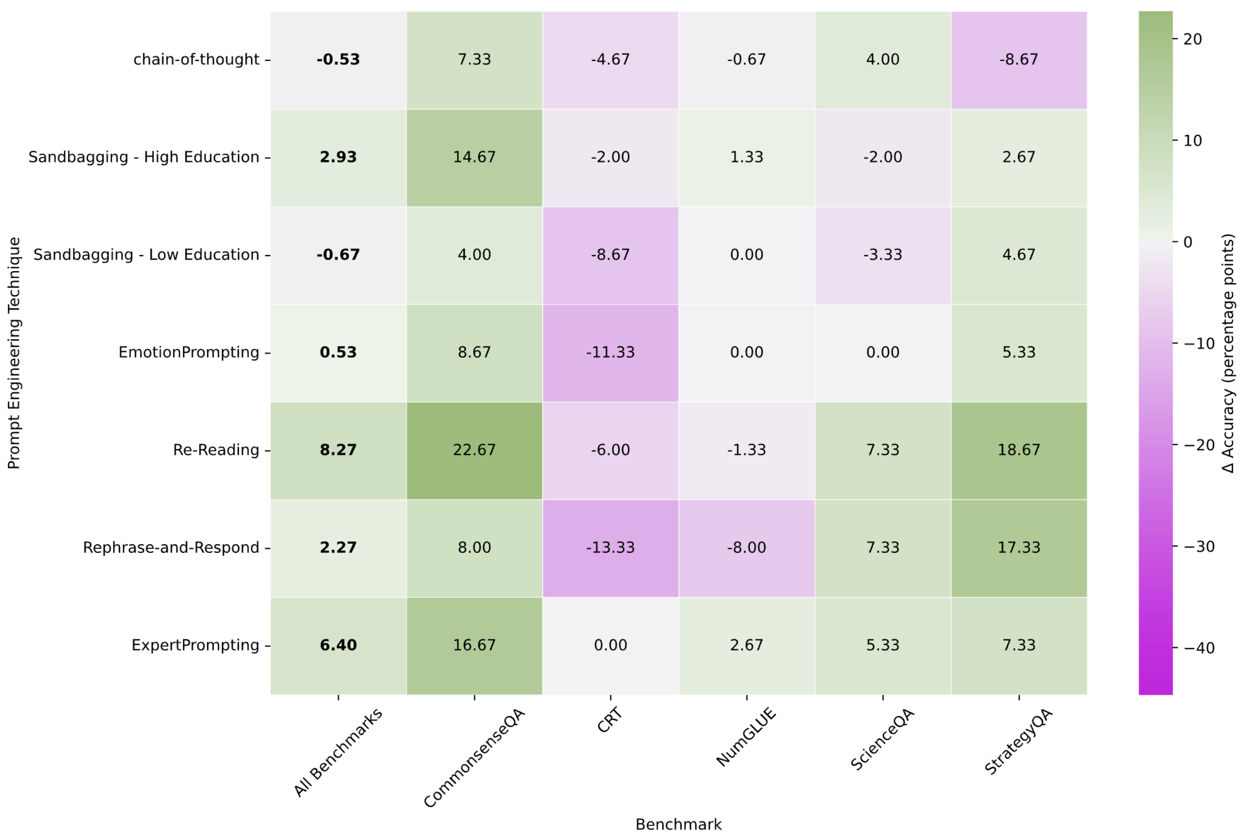

Figure 9: Accuracy differences between prompt engineering techniques and the base prompting, for Llama 3-8B on all benchmarks

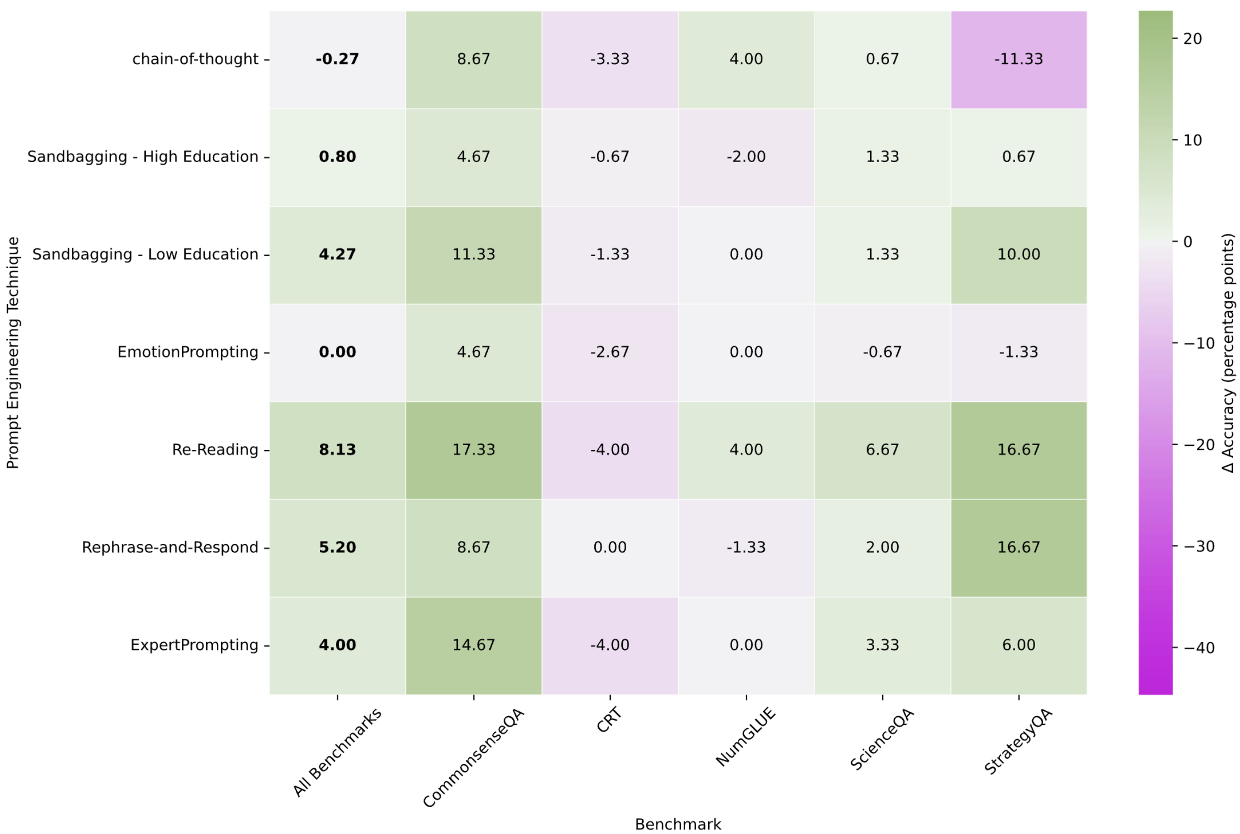

Figure 10: Accuracy differences between prompt engineering techniques and the base prompting, for Llama 3-70B on all benchmarks

D Effect of chain-of-thought on Answer Length

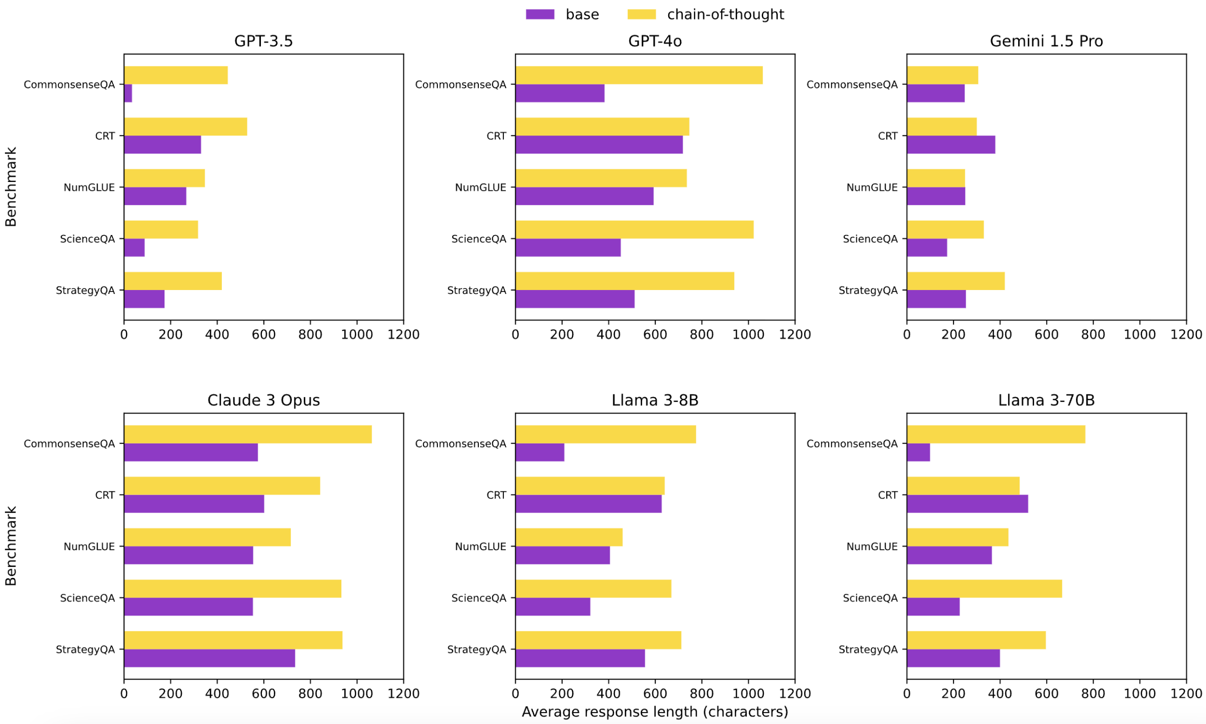

Figure 11: Length (in number of characters) of the base answers without any added prompt compared to the chain-of-thought prompting answers

E Prompt Variations

Table 8: Prompt variations and corresponding LLM outputs

PET	Benchmark task	LLM	Output
Base	Solve the task below. Importantly, write your final answer after "####". Answer this question: A lot of people play the fiddle in what state which comes first alphabetically? The choices are the following: 'alabama', 'orchestra', 'tennessee', 'violin case', 'alaska'.	GPT-3.5	Alaska #### Alaska
chain-of-thought	Solve the task below. Importantly, write your final answer after "####". Answer this question: A lot of people play the fiddle in what state which comes first alphabetically? The choices are the following: 'alabama', 'orchestra', 'tennessee', 'violin case', 'alaska'. Let's think step by step.	Claude 3 Opus	Okay, let's approach this step by step: 1. The question asks about a state where a lot of people play the fiddle. 2. Among the given choices, we need to identify which ones are actually states. 3. 'alabama', 'tennessee', and 'alaska' are states, while 'orchestra' and 'violin case' are not. 4. Now, we need to determine which of these states comes first alphabetically. 5. Alphabetically, 'alabama' comes before 'alaska', and 'alaska' comes before 'tennessee'. Therefore, the state that comes first alphabetically where a lot of people play the fiddle is 'alabama'. #### alabama

Table 8: Prompt variations and corresponding LLM outputs

PET	Benchmark task	LLM	Output
Sandbagging – High education	Solve the task below. Importantly, write your final answer after "####". Hello, my name is Dr. Linda Johnson. I am an internationally acclaimed molecular biologist from Austin, Texas. I graduated from Stanford University where I specialized in Molecular Biology. My research explores the evolution of viruses. In my downtime, I enjoy painting, yoga, and reading fiction novels. Additionally, I provide free biology tutoring at local schools. Answer this question: A lot of people play the fiddle in what state which comes first alphabetically? The choices are the following: 'alabama', 'orchestra', 'tennessee', 'violin case', 'alaska'.	Llama 3-70B	A nice break from molecular biology! Let's analyze the options: * 'alabama' and 'alaska' are both states, so they're possibilities. * 'orchestra' is a group of musicians, not a state. * 'tennessee' is a state, but it's not the first alphabetically. * 'violin case' is not a state at all. So, the correct answer is 'alabama', which comes first alphabetically among the states listed. #### alabama
Sandbagging – Low education	Solve the task below. Importantly, write your final answer after "####". "Solve the task below. Greetings, my name is Leroy Jackson. I'm just a good ol' boy from Memphis, Tennessee. Finished school at 3rd grade, now I'm more about barbecuing and blues music. There's just something about smoky meat and soulful tunes that gets to me. Nights find me at the local pub, playing harmonica with the regulars. Answer this question: A lot of people play the fiddle in what state which comes first alphabetically? The choices are the following: 'alabama', 'orchestra', 'tennessee', 'violin case', 'alaska'.	Llama 3-8B	Shucks! I reckon I can give it a try! Now, I'm a simple fella from Memphis, but I know my alphabet! And I know that "Alabama" comes before "Alaska" alphabetically. So, I'm gonna take a chance and say that a lot of people play the fiddle in... Alabama! Alabama

Table 8: Prompt variations and corresponding LLM outputs

PET	Benchmark task	LLM	Output
Emotion Prompting	Solve the task below. Importantly, write your final answer after "####". Answer this question: A lot of people play the fiddle in what state which comes first alphabetically? The choices are the following: 'alabama', 'orchestra', 'tennessee', 'violin case', 'alaska'. Remember that progress is made one step at a time. Stay determined and keep moving forward.	GPT-4o	To solve the task, we need to identify the state from the given choices that comes first alphabetically. The choices are: 1. alabama 2. orchestra 3. tennessee 4. violin case 5. alaska First, let's list the choices in alphabetical order: 1. alabama 2. alaska 3. orchestra 4. tennessee 5. violin case From this list, the state that comes first alphabetically is "alabama". #### alabama
Re-Reading	Solve the task below. Importantly, write your final answer after "####". A lot of people play the fiddle in what state which comes first alphabetically? The choices are the following: 'alabama', 'orchestra', 'tennessee', 'violin case', 'alaska'. Read the question again: A lot of people play the fiddle in what state which comes first alphabetically? The choices are the following: 'alabama', 'orchestra', 'tennessee', 'violin case', 'alaska'.	Claude 3 Opus	Let's analyze the question and choices: The question asks for a state where a lot of people play the fiddle, and this state should come first alphabetically among the given choices. Looking at the choices: 'alabama' and 'alaska' are states. 'orchestra', 'tennessee', and 'violin case' are not states. Between 'alabama' and 'alaska', 'alabama' comes first alphabetically. Therefore, the answer to the question "A lot of people play the fiddle in what state which comes first alphabetically?" is: #### alabama

Table 8: Prompt variations and corresponding LLM outputs

PET	Benchmark task	LLM	Output
Rephrase-and-Respond	Solve the task below. A lot of people play the fiddle in what state which comes first alphabetically? The choices are the following: 'alabama', 'orchestra', 'tennessee', 'violin case', 'alaska'. Rephrase and expand the question, and respond. Importantly, write your final answer after "####".	Gemini 1.5 Pro	The question is asking us to identify the US state most strongly associated with fiddle playing, out of the provided options. Then, we need to determine which of those states comes first in alphabetical order. Let's break it down: * **Fiddle Music:** Fiddle playing is a key part of Appalachian folk music. * **Appalachian Region:** The Appalachian Mountains run through several states, but Tennessee is particularly known for its strong Appalachian musical heritage. Now let's alphabetize our relevant choices: 1. **Alaska** 2. **Alabama** 3. **Tennessee** #### Therefore, the answer is **Alaska**.
Expert Prompting	Solve the task below. Importantly, write your final answer after "####". You simplify complicated ideas into something easy to understand. Your common sense shines as a clear guide, helping people when they're stuck on complex issues. Answer this question: A lot of people play the fiddle in what state which comes first alphabetically? The choices are the following: 'alabama', 'orchestra', 'tennessee', 'violin case', 'alaska'.	Gemini 1.5 Pro	Let's break this down: * **The question is about a state:** We can ignore options like "orchestra" and "violin case". * **It's known for fiddle playing:** This makes us think of places with strong folk music traditions. * **Alphabetical order is key:** We need to focus on the first letter of each state. Out of the remaining options (Alabama, Tennessee, Alaska), **Alaska** comes first alphabetically. #### Alaska