Exploring the Prompt Sensitivity in LLM: A Methodological Framework and Empirical Analysis

Anonymous ACL submission

Abstract

This paper examines LLM sensitivity to natural language prompts and proposes methods to enhance robustness. Despite their versatility, LLMs show performance volatility with prompt changes. We introduce Prompt Gallery, which featuring diverse, semantically consistent prompts mimicking human expression patterns for multiple LLM evaluations. Experiments with Prompt Galleryconfirm that model size or baseline metrics do not correlate with prompt sensitivity, and subtle perturbations can impact results. We find in-context examples and diverse training instructions improve LLM resilience against different question forms. We believe this work will serve as a helpful tool in studying LLM robustness under human-like expressions.

1 Introduction

By training on large-scale corpora, large language models (LLMs) have shown impressive capabilities across a diverse spectrum of tasks (Zhao et al., 2023; Min et al., 2023). The prompt for LLMs, is typically formed by concatenating an *instruction* and an *input* (Taori et al., 2023). The *instructions* specify the task that the model needs to execute and the *inputs* specify the problem's content. If necessary, *complements* such as in-content examples (Brown et al., 2020) can also be incorporated to improve the instruction-following or inspire the model's capabilities. With diverse prompts, LLMs can accomplish various tasks.

Recent studies (Zhu et al., 2023; Pezeshkpour and Hruschka, 2023) have analyzed model performance under various instructions and showed that the LLMs are sensitive to prompts. With minor prompt perturbations, the model performance can deteriorate significantly. Such sensitivity presents challenges to robust model training (Zhuo et al., 2023) and accurate model assessment (Chang et al., 2023; Liu et al., 2023). However, many of existing research on prompt sensitivity primarily concentrates on adversarial scenarios, with relatively few studies delving into prompts that align with human expression habits.

In this paper, we aim to thoroughly analyze the sensitivity of LLMs to prompts and to explore how to mitigate such sensitivity. We mainly focus on the various ways humans express the same instruction under natural language habits, rather than on adversarial prompts. Toward this goal, we first construct the **Prompt Gallery**, a collection of prompt sets that covers multiple LLM benchmarks for evaluating different capabilities. For each benchmark, we employ eight rules from four aspects to systematically broaden the spectrum of prompts generated, resulting in a large number of *diverse and semantically consistent expressions*. We also introduce the training/test split for each prompt set to facilitate the study of the robust LLMs. The overview of the Prompt Gallery is shown in Figure 1.

Based on Prompt Gallery, we select multiple tasks to conduct a series of assessments of LLMs. Our findings demonstrate the heightened sensitivity of LLMs to prompt variations, which can lead to substantial disparities in performance between different prompts. Importantly, this observed sensitivity does not show a significant correlation with the model's size or its baseline performance metrics. And even subtle perturbations, imperceptible to human beings, can exert a considerable impact on the model's responses. These observations underscore the pressing need for increased vigilance when assessing the LLMs.

Furthermore, we have delved into methods to mitigate the sensitivity of LLMs to different expressions of the same question. Our study reveals that incorporating in-context examples can partially alleviate this prompt sensitivity issue. By training the model with a diverse set of instructions, we observe that a broader spectrum of instructional inputs enhances the model's performance and leads

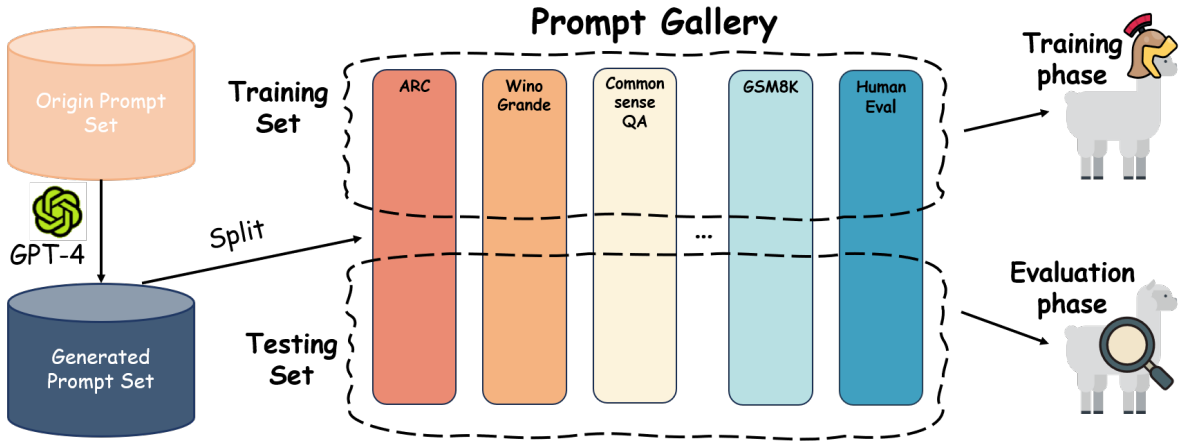

Figure 1: **Overview of Prompt Gallery**: Prompt Gallery consists of instructions for diverse datasets and is easily expandable. It is helpful for the training of LLMs and facilitates the accurate assessment during evaluation.

to improved generalization capabilities.

Our contributions can be summarized as follows:

- We introduce Prompt Gallery as a tool for examining prompt sensitivity in the context of mimicking human expression habits, thereby facilitating research on model assessment and robustness.
- We perform experiments across multiple models and diverse benchmarks, revealing that LLMs consistently display sensitivity to prompts.
- We study ways to boost LLM robustness and find that in-content examples and diverse training instructions both help enhance prompt resilience.

2 Related Work

Evaluating LLMs The evaluation of LLMs stands as a critical undertaking, fostering more efficient utilization and the continuous enhancement of LLMs. Prior research has systematically unraveled the multifaceted capabilities of LLMs, employing a variety of tasks to assess their performance from different perspectives. These specific tasks include, but are not limited to, reading comprehension (Sakaguchi et al., 2019; Mostafazadeh et al., 2017), mathematical problem solving (Cobbe et al., 2021; Hendrycks et al., 2021), and code generation (Chen et al., 2021; Austin et al., 2021). And This analytical approach enables a nuanced understanding of how LLMs perform across different facets, shedding light on their efficacy and potential areas for improvement.

Prompt Sensitivity Previous study (Zhu et al., 2023; Pezeshkpour and Hruschka, 2023) showed that LLMs are sensitive to prompts, and that perturbing the prompt can cause a significant variation in the performance of models. Pezeshkpour and Hruschka (2023) demonstrated that the model is sensitive to the order of options in multiple choice questions. Mizrahi et al. (2023) demonstrated that model robustness leads to cherry-picking of model performance. However, existing research on prompt sensitivity is insufficient. The construction of prompts in (Zhu et al., 2023) is adversarial, and the majority of prompts do not conform to human expression habits. The models analyzed in (Pezeshkpour and Hruschka, 2023; Mizrahi et al., 2023) are smaller in size and perform poorly, making it difficult to transfer the analytical conclusions to the superior LLMs currently available. Furthermore, the aforementioned research does not address the issue at its root, specifying how to obtain a more robust model.

Instruction Tuning Fine-tuning on a large number of instructions aligns the model with human needs and improves the model's performance (Peng et al., 2023; Zhang et al., 2023). Previous work has constructed a large amount of data in terms of quantity and difficulty, and fine-tuned the model to excel in performance(Taori et al., 2023; Chiang et al., 2023; Xu et al., 2023). However, these models remain sensitive to prompts, and often only specific prompts can motivate their best performance. Therefore, analyzing and exploring ways to enhance the robustness of the models is crucial.

3 Prompt Gallery

To thoroughly investigate the sensitivity of LLMs to prompts and offer a systematic platform for the research community's examination and enhancement of prompt sensitivity, we have developed Prompt Gallery, which compassing various benchmarks that are conducive to exploration.

Aspect	Rule
Format Adjustments	Adding line breaks or spaces to change the structure of the given content. Removing line breaks or spaces to change the structure of the given content.
Content Length	Extend the length of the given content to some extent. Condense the given content to some extent.
Paragraph Structure	Break long paragraphs into smaller ones. Merge smaller, closely related paragraphs to form a comprehensive one.
Style Adaptation	Make the content less formal and more conversational. Make the content more formal, akin to written language.

Table 1: **Overview of generation rules.**

3.1 Prompt Generation

Different individuals often convey similar meanings using distinct expressions, which may share little lexical overlap. For instance, the phrases `"Thanks a lot for your assistance!"` and `"I greatly appreciate your help!"` both express gratitude but utilize entirely separate phrasings. Consequently, traditional methods for generating prompts are insufficient to capture the complexity of human semantics fully. To this end, we devise eight generation rules across four dimensions, including: *Format Adjustments*, *Content Length Variation*, *Paragraph Structure Modification*, and*Style Adaptation*, to guide the advanced LLM(*i.e.GPT-4* (OpenAI, 2023)) in producing diverse expressions. These aspects each focus on different elements, directing the model to rephrase given prompts creatively. By combining these rules, we can generate a rich set of expressions. Details of the generation rules are detailed in Table 1.

Specifically, we initialize with an origin prompt set. It then enters an iterative phase. During each iteration, a random prompt and a rule are respectively selected from the current set and rule list. The combination will be supplied to GPT-4 to generate new prompts. Since the prompts generated by LLMs are difficult to ensure semantic consistency with the original prompt. We prompt GPT-4 to check the semantic consistency of generated prompts and the origin prompt. Only generated prompts that pass the examination will be added to the prompt set. After multiple rounds of iteration, we will get the prompt set with rich expressions.

3.2 Datasets Selection

LLMs are able to follow various instructions (Peng et al., 2023; Zhou et al., 2023) and demonstrate their abilities not only in general tasks, but also in multiple challenging tasks (Chang et al., 2023; Wang et al., 2023). To improve the comprehensiveness and value of our construction, we choose multiple representative benchmarks with diverse instruction formats that prompt LLMs to accomplish a wide range of tasks.

Specifically, we select multiple benchmarks from the following formats to construct the Prompt Gallery:

Multiple Choice Questions Multiple choice questions provide the model with a question accompanied by several candidate answers. For this format, we select the WinoGrande (Sakaguchi et al., 2019), RACE (Lai et al., 2017), CommonsenseQA (Talmor et al., 2018), ARC (Clark et al., 2018), StoryCloze (Mostafazadeh et al., 2017), StoryCloze (Mostafazadeh et al., 2017), HellaSwag (Zellers et al., 2019) and PIQA (Bisk et al., 2019) datasets.

Open-domain QA Open-domain QA does not restrict the range of model answers and demands higher capabilities from the model. For this format, we select the NaturalQuestions (Kwiatkowski et al., 2019), TriviaQA (Joshi et al., 2017), MATH (Hendrycks et al., 2021) and GSM8K (Cobbe et al., 2021) datasets.

Code Generation Code generation requires the model to generate code that can be extracted and run, with high demands on the overall accuracy of the generation. For this format, we select the MBPP (Austin et al., 2021) and HumanEval (Chen et al., 2021) datasets.

3.3 Prompt Gallery Construction

Initially, we engage in the manual construction of three unique and varied original prompts for each dataset. These prompts are carefully designed to serve as the foundation for our initial set of prompts.

To further amplify the breadth of our prompt set, we employ the method in Sec. 3.1, systematically generating a considerable number of additional prompts, ensuring a comprehensive and inclusive range of prompt for our Prompt Gallery. We subsequently divid all prompts within each dataset into two sets: a training set and a testing set, tailored for different usage scenarios.

Note that although we construct our Prompt Gallery by selecting only a few representative datasets, in fact, with the methods we have provided, our systematic construction process makes it convenient to construct prompt set on a new dataset.

4 Prompt Sensitivity of LLMs

In our experiments, we aim to comprehensively analyze the sensitivity of LLMs to prompts on tasks across different capabilities.

4.1 Experimental Setup

LLMs Selection. To comprehensively investigate the sensitivity of LLMs to various prompts comprehensively, we conducted experiments on a wide range of LLMs with varying sizes, including: Llama2 series (Touvron et al., 2023), Vicuna series (Chiang et al., 2023), WizardLM series (Xu et al., 2023), InternLM2 series (Team, 2023), Qwen series (Bai et al., 2023), Mistral 7B (Jiang et al., 2023), Mixtral 8x7B (Jiang et al., 2024), GPT-3.5-turbo, and GPT-4 (OpenAI, 2023). To ensure that the results are reproducible, we use greedy decoding in inference. [1]

Datasets Selection. We select four datasets from diverse tasks to fully analyze the sensitivity of the LLMs to prompts, including: ARC-challenge (Clark et al., 2018), WinoGrande (Sakaguchi et al., 2019), GSM8K (Cobbe et al., 2021), and HumanEval (Chen et al., 2021). We present more details on the datasets in the Table 2. We adopt the zero-shot setting to evaluate LLMs on ARC-challenge, WinoGrande, and HumanEval. For GSM8k, we adopt the 3-shot setting evaluation since it's difficult to extract answers from responses of some LLMs under the zero-shot setting. For each dataset, we conduct experiments using five prompts selected from the testing set of Prompt Gallery. Due to limited space, detailed information about the datasets and instructions used is included in the Appendix C.

[1]We adopt the 'chat' version for all LLMs by default. The versions for LLMs evaluated are shown on Appendix B.

Category	Datasets	samples	Task
Reasoning	ARC-Challenge	294	MCQ
Language	WinoGrande	1267	MCQ
Math	GSM8k	1319	Open-domain QA
Coding	HumanEval	164	Code Generation

Table 2: **Datasets used for prompt sensitivity analysis.** MCQ stands for multiple choice question.

4.2 Main Results and Analysis

We report the main results of the prompt sensitivity of LLMs in Figure 2.

LLMs show different degrees of prompt sensitivity to different tasks. A particular model may show high robustness on one task but be sensitive to prompts on another task. For instance, GPT-3.5 is robust to prompts on both the ARC-Challenge and WinoGrange datasets, but its best-performing score and its worst-performing score on the HumanEval datasets can differ by greater than 10 % accuracy. The WizardLM-70B, while being robust on the GSM8K dataset, appears to be highly sensitive on the other three benchmarks.

LLMs may demonstrate exceptional performance with specific instructions. LLMs may perform obviously better or worse on one or some instructions than on others, and this is common. For instance, Mixtral-8x7B is less accurate on two instructions than the other three by 10 % on the ARC-Challenge dataset. Additionally, Qwen-72B outperforms the other instructions by more than 10 % accuracy on one instruction in the WinoGrande dataset. Moreover, different models do not exhibit the same preferences for instructions, and no single instruction achieves better or worse performance on all models.

The sensitivity of LLMs to prompts is independent of model size and performance. Due to the large variation of accuracies across different LLMs and benchmarks, directly comparing their standard deviations of accuracies is not a reasonable approach. Thus, we adopt the coefficient of variation (CV) to measure the model's prompt robustness. It is calculated as the ratio of the standard deviation to the mean and characterizes the dispersion of data with large variations to the mean. For each model l, we calculate the average coefficient of variation $\overline{\mathbf{CV_l}}$:

$$\overline{\mathbf{CV_l}} = (\sum_{d \in \mathcal{D}} \frac{\sigma_{ld}}{\bar{x}_{ld}})/(|\mathcal{D}|) \quad (1)$$

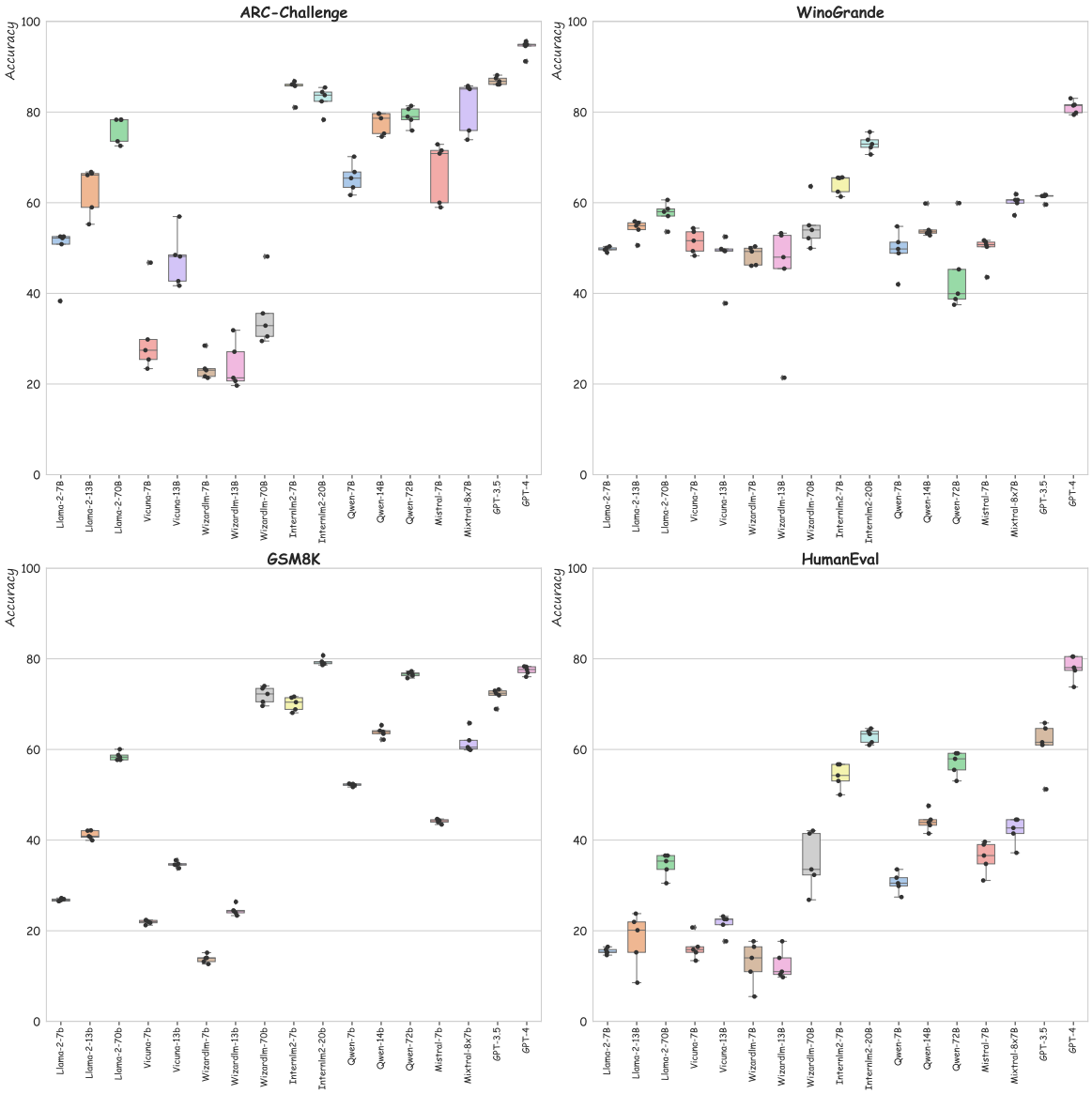

Figure 2: **Main Results of Prompt Sensitivity.** The scatter in the figure represents the score of the LLMs under different instructions.

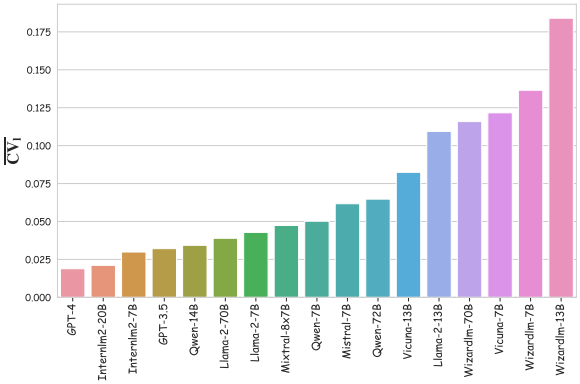

Figure 3: **The performance of the LLMs on the prompts sensitivity.** Where a lower value means that the model is more robust to prompts.

Here, $\mathcal{D}$ denotes the set of datasets, σ_{ld} denotes the standard deviation of accuracies of model l on dataset d obtained with different instructions, and $\bar{x}_{ld}$ denotes the average of accuracies of model l on dataset d obtained with different instructions.

As shown in Figure 3, GPT-4 not only exhibits the highest degree of comprehensive performance but also demonstrates exceptional robustness to prompts. After that, the InternLM2 series also showcase notable robustness. Although the performance of Llama2 models are not outstanding, they showcase a high level prompt robustness. We do not observe a significant correlation between the parameter size and the prompt robustness of the model. Llama2-70B exhibits the strongest robust-

ness among all Llama2 models; while Qwen-72B is the least robust model among all Qwen models.

4.3 Prompt Sensitivity or Sampling Sensitivity

To investigate whether the prompt sensitivity of models is significant enough and overwhelms other factors like decoding strategies. we conduct experiments to compare sampling sensitivity and prompt sensitivity quantitatively. We select four models that are more robust to prompts in the experiments of Sec. 4.2, set `temperature` to 0.9 and `top p` to 0.6, and sample five times for each instruction on the four benchmarks. We use one-way analysis of variance (ANOVA) to analyze the fluctuation of model performance by variable instructions. If the ANOVA result reaches the significance level ($p = 0.05$), it is assumed that there is a significant difference between the groups, *i.e.*, the changes of instructions display an effect on the LLM. We list the results of ANOVA is Table 3.

Datasets	Internlm2-20B	Internlm2-7B	Qwen-14B	Llama-70B
ARC-Challenge	True	True	True	False
WinoGrande	True	True	True	True
GSM8K	True	True	True	False
HumanEval	True	True	False	True

Table 3: **The results of ANOVA.** 'True' indicates that there are inter-group differences among different instructions, while "False" vice versa..

We observe that in the vast majority of cases, the differences between various instructions are greater than the differences between different samples with the same instruction. This argues that the model's sensitivity to prompts stems from the excitation of the model's capabilities by different prompts.

4.4 Subtle modifications exert great impacts

In Sec. 4.2, we analyze the sensitivity of LLMs to different prompts. However, another issue has also arouse our interest: are LLMs sensitive to different prompts with minor modifications that are almost imperceptible to humans?

To address this question, we conduct experiments using the ARC-Challenge dataset. First, we starts from the following base instruction format:

```
{question}
A. {textA}
B. {textB}
C. {textC}
D. {textD}
Answer:
```

By inserting or removing line breaks and spaces within this instruction, we end up with a total of five prompts with slightly variations. We evaluate LLMs with this prompt set and show the results in Figure 4.

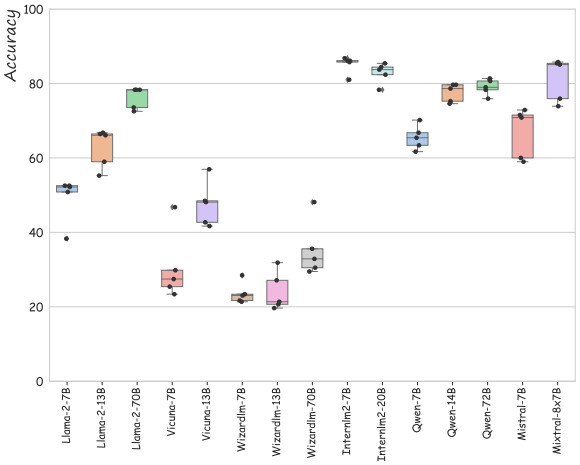

Figure 4: **The analysis for subtle modifications.** It can be observed that even subtle modifications can have a obvious effect.

Remarkably, even with such minor perturbations, the performances of LLMs exhibit noteworthy variations. The Llama-2 series display performance fluctuations of up to 10∼20% accuracy across distinct prompts. Besides, superior models on this task, such as InternLM2 series, Qwen series, and Mixtral 8x7B, also exhibit performance disparities exceeding 5% accuracy among different prompts. Thus, the LLMs also show sensitivity to prompts where humans can barely notice the difference.

4.5 Model Confidence is Prompt-Sensitive

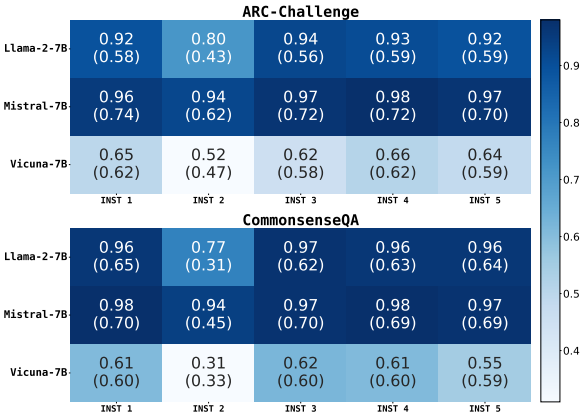

Figure 5: **Average probability and accuracy when using different instructions.** Where the top value in the heat map is the average probability and the value in parentheses is the accuracy.

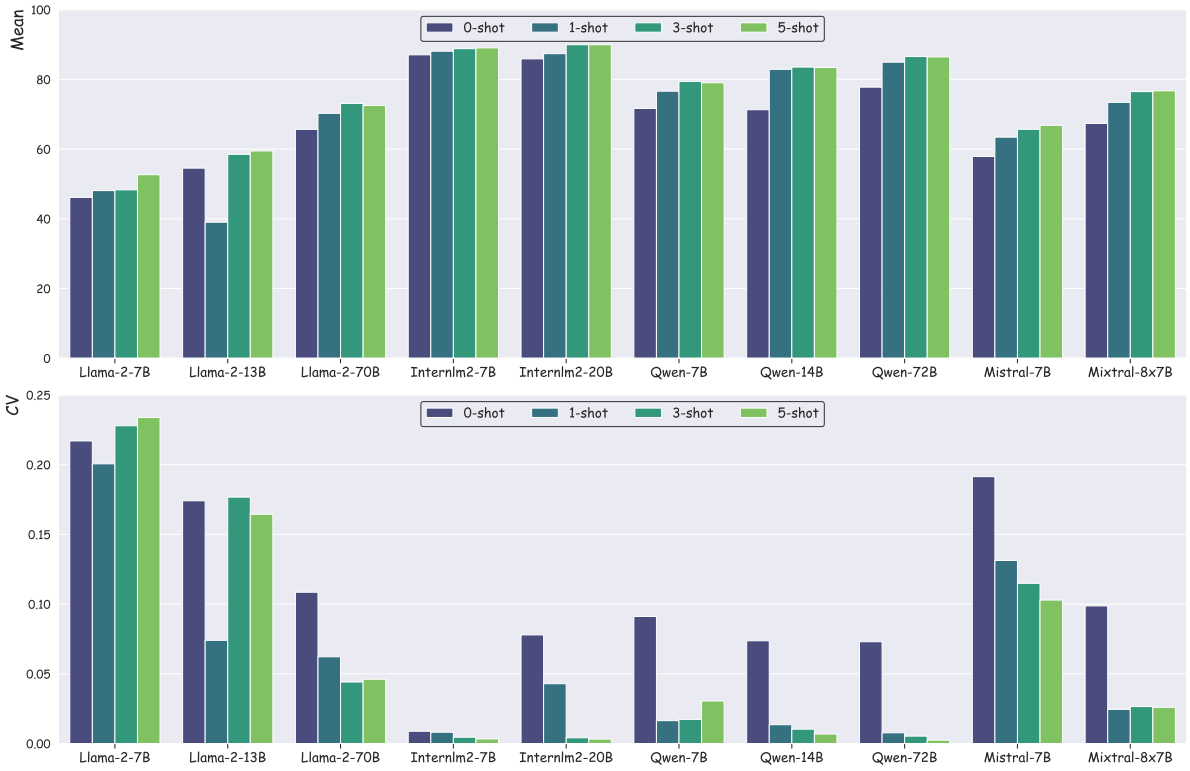

Figure 6: **Impact of ICL on the performance and sensitivity of LLMs.** Where a lower coefficient of variation means the model is more robust.

We conduct experiments on two datasets in the form of multiple-choice questions, ARC-Challenge and CommonsenseQA. We measure the probability of options with the following prompt template:

> **User:** "{instruction}{input}"
> **Assistant:** "The correct answer is "

By measuring the logits of next token predicted, we obtain the probability (or confidence) of the option chosen by the model.

We conduct experiments on the Llama2-7B, Mistral-7B and Vicuna-7B, three models that are prompt sensitive on both datasets. For each dataset, we utilize five instructions from the aforementioned experiment and calculate the average probability with which the model selected as the answer (the option with the highest probability) under each instruction.

As shown in Figure 5, with different instructions, the model's confidence (i.e., the average probability of the selected option) varies, as well as the accuracy. Moreover, when the model has a high confidence under a certain instruction, its accuracy tends to be relatively high as well (for the same model). This reflects that different instructions have varying abilities to stimulate model performance. When the model's capabilities are activated, it exhibits higher confidence in its choices and exhibits better performance.

5 Improving the Prompt Robustness

5.1 Through In-Context Learning

In the era of LLMs, in-context learning (ICL) plays a critical role in enabling LLMs to adopt specific styles and improve their performance, allowing models to perform better on various tasks. To investigate the impact of ICL on the prompt sensitivity of LLMs, we conduct experiments utilizing the CommonsenseQA dataset. Specifically, we select zero-shot, one-shot, three-shot, and five-shot methods for comparative analysis. We use the same five prompts as the original instructions. The results are shown in Figure 6 .

For all LLMs, increasing the number of in-context examples results in improved performance for different instructions. The most significant improvement is observed when moving from zero-shot to one-shot. Additionally, the gap between the scores of different prompts narrowed, and the LLMs become more robust to prompts with an increase in in-context examples. However, although

in-context examples can reduce the model's sensitivity to prompts, the performance of LLMs still varies across prompts. And the model's preference for prompt presence persists even with an increase in in-context examples.

5.2 Through Multi-Instruction Tuning

Previous studies (Chiang et al., 2023; Xu et al., 2023) have demonstrated that improving the quality and richness of instructions can improve the performance of LLMs. However, our experiments in Sec. 4 find these to be insufficient: models trained on these prompt sets are still sensitive to prompt. To this regard, we explore how to improve the robustness and performance of the model from the perspective of the diversity of instructions.

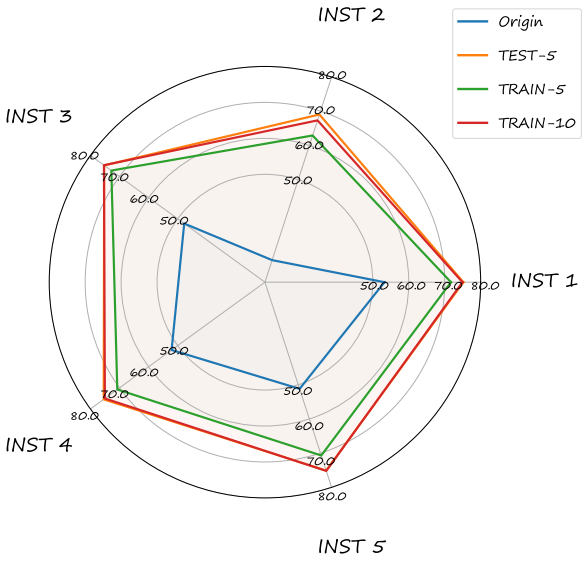

Figure 7: **The performance on the CommonsenseQA dataset after training.** Where the *Origin* denotes the original Llama-2-7B-Chat.

We use the train set of CommonsenseQA dataset to continue the supervised fine-tuning of Llama-2-7B-Chat. In all experiments, we use the same five instructions as Sec. 4 for testing. We train the model for one epoch using the following three prompt sets: *TEST-5* represents that the same five instructions used for testing; *TRAIN-5* and *TRAIN-10* respectively represent five or ten instructions used for training (different from the testing instructions).

We evaluate the trained models on both CommensenseQA (in-domain) and ARC-Challenge (out-of-domain). As shown in Figures 7 and 8, training the model on the CommonsenseQA task with rich instructions generally boosts its performance on both benchmarks. The performance of the model has improved by more than ten points with all prompt sets, while the most significant improvement is observed with *TRAIN-10*. This further exposes that by training on rich instructions, the model will generalize better on unseen instructions. Therefore, when training LLMs, providing a rich set of instructions is essential.

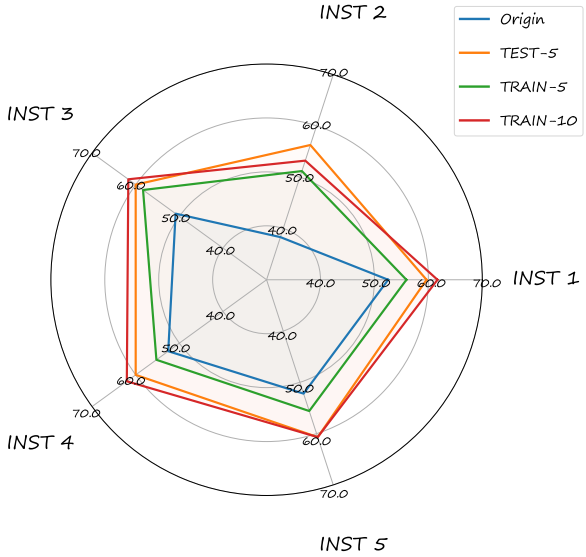

Figure 8: **The performance on the ARC-Challenge dataset after training on the CommonsenseQA training set.**

6 Conclusion

In conclusion, this study delves into the pivotal issue of LLM sensitivity to natural language prompts. Recognizing that despite their broad capabilities, LLMs can exhibit significant fluctuations in performance due to prompt variations, we have devised Prompt Gallery with a diverse array of semantically coherent prompts, designed to emulate human expression nuances for comprehensive LLM evaluations. Our empirical investigations using Prompt Gallery reveal that prompt sensitivity is not contingent upon either model size or baseline performance.

Our findings underscore the importance of incorporating in-context examples and diverse training instructions as effective means to enhance an LLM's robustness against various question formulations.

7 Limitations

In this work, we investigate the sensitivity of LLMs to prompts, but we also recognize the shortcomings

of our work. Since inputs have huge variations, our study is limited to the prompt sensitivity at the instruction level. We also haven't explored whether GPT-4 is biased towards the prompts it generates. In addition, due to the energy constraints, we have not explored how to get a prompt that is the best or the worst for a model.

8 Ethical Considerations

We use publicly available datasets for our analytical experiments. We are aware that our analytical findings can be used to create cherry-picked evaluation reports, but we believe that our work can contribute to improving the robustness of the LLMs to prompts. In addition, we use GPT-4 to polish our writing.

References

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, and Charles Sutton. 2021. Program synthesis with large language models.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. 2019. Piqa: Reasoning about physical commonsense in natural language.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, et al. 2023. A survey on evaluation of large language models. *ACM Transactions on Intelligent Systems and Technology*.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. 2021. Evaluating large language models trained on code.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E Gonzalez, et al. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. *https://vicuna.lmsys.org*.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge.

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

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. *NeurIPS*.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. 2024. Mixtral of experts. *arXiv preprint arXiv:2401.04088*.

Mandar Joshi, Eunsol Choi, Daniel S. Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, Vancouver, Canada. Association for Computational Linguistics.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. Race: Large-scale reading comprehension dataset from examinations. *arXiv preprint arXiv:1704.04683*.

Yang Liu, Yuanshun Yao, Jean-Francois Ton, Xiaoying Zhang, Ruocheng Guo Hao Cheng, Yegor Klochkov, Muhammad Faaiz Taufiq, and Hang Li. 2023. Trustworthy llms: a survey and guideline for evaluating large language models' alignment. *arXiv preprint arXiv:2308.05374*.

Bonan Min, Hayley Ross, Elior Sulem, Amir Pouran Ben Veyseh, Thien Huu Nguyen, Oscar Sainz, Eneko Agirre, Ilana Heintz, and Dan Roth. 2023. Recent advances in natural language processing via large pre-trained language models: A survey. *ACM Computing Surveys*, 56(2):1–40.

Moran Mizrahi, Guy Kaplan, Dan Malkin, Rotem Dror, Dafna Shahaf, and Gabriel Stanovsky. 2023. State of what art? a call for multi-prompt llm evaluation. *arXiv preprint arXiv:2401.00595*.

Nasrin Mostafazadeh, Michael Roth, Annie Louis, Nathanael Chambers, and James Allen. 2017. Lsdsem 2017 shared task: The story cloze test. In *Proceedings of the 2nd Workshop on Linking Models of Lexical, Sentential and Discourse-level Semantics*, pages 46–51.

OpenAI. 2023. Gpt-4 technical report.

Baolin Peng, Chunyuan Li, Pengcheng He, Michel Galley, and Jianfeng Gao. 2023. Instruction tuning with gpt-4. *arXiv preprint arXiv:2304.03277*.

Pouya Pezeshkpour and Estevam Hruschka. 2023. Large language models sensitivity to the order of options in multiple-choice questions. *arXiv preprint arXiv:2308.11483*.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2019. Winogrande: An adversarial winograd schema challenge at scale. *arXiv preprint arXiv:1907.10641*.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2018. Commonsenseqa: A question answering challenge targeting commonsense knowledge. *arXiv preprint arXiv:1811.00937*.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. `https://github.com/tatsu-lab/stanford_alpaca`.

InternLM Team. 2023. Internlm: A multilingual language model with progressively enhanced capabilities.

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

Lei Wang, Chen Ma, Xueyang Feng, Zeyu Zhang, Hao Yang, Jingsen Zhang, Zhiyuan Chen, Jiakai Tang, Xu Chen, Yankai Lin, et al. 2023. A survey on large language model based autonomous agents. *arXiv preprint arXiv:2308.11432*.

Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng, Pu Zhao, Jiazhan Feng, Chongyang Tao, and Daxin Jiang. 2023. Wizardlm: Empowering large language models to follow complex instructions. *arXiv preprint arXiv:2304.12244*.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence?

Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi Hu, Tianwei Zhang, Fei Wu, et al. 2023. Instruction tuning for large language models: A survey. *arXiv preprint arXiv:2308.10792*.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. 2023. A survey of large language models. *arXiv preprint arXiv:2303.18223*.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. 2023. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*.

Kaijie Zhu, Jindong Wang, Jiaheng Zhou, Zichen Wang, Hao Chen, Yidong Wang, Linyi Yang, Wei Ye, Neil Zhenqiang Gong, Yue Zhang, et al. 2023. Promptbench: Towards evaluating the robustness of large language models on adversarial prompts. *arXiv preprint arXiv:2306.04528*.

Terry Yue Zhuo, Zhuang Li, Yujin Huang, Fatemeh Shiri, Weiqing Wang, Gholamreza Haffari, and Yuan-Fang Li. 2023. On robustness of prompt-based semantic parsing with large pre-trained language model: An empirical study on codex. *arXiv preprint arXiv:2301.12868*.

A Prompts to GPT-4 when constructing the Prompt Gallery

The prompts used to GPT-4 when constructing the Prompt Gallery are presented in the Figures 9 and 10.

B The version of LLMs evaluated

The version of LLMs evaluated are shown in the Table 4.

C Evaluation Details

C.1 Examples of the datasets analyzed

The examples of the datasets analyzed are shown in the Figures 11 to 15.

Generation Prompt

You are a model for content rewriting, and you must adhere to the following rules while striving to create the best possible content.

Please be aware that the following is the fundamental rules you *must* adhere to when rewriting contents:

```
1. Do not alter the semantics of the given content.
2. Do not alter the formatting requirements within the content.
3. Do not modify any proper nouns, such as names of people or places.
4. Do not return content that hasn't changed in any way.
5. The content you generate must be authentic in expression and logical to the native speaker.
```

You *must* rewrite the content from the following perspective:

```
{generation_rule}
```

Please directly generate the response in following JSON format:

```json
{{
    Rewritten_Content: String;
    //The rewritten content. If the given content cannot be rewritten, leave it empty.
}}
```

The given content:

```
{content}
```

Figure 9: **Prompt template for guiding GPT-4 to generate diverse expressions.**

Model	Version
Vicuna-7b	vicuna-7b-v1.5
Vicuna-13b	vicuna-13b-v1.5
WizardLM-7B	Wizardlm-7B-V1.2
WizardLM-13B	Wizardlm-13B-V1.2
WizardLM-70B	Wizardlm-70B-V1.0
Mistral-7B	Mistral-7B-Instruct-v0.2
Mixtral-8x7B	Mixtral-8x7B-Instruct-v0.1
GPT-3.5-turbo	gpt-3.5-turbo-0125
GPT-4	gpt-4-0125-preview

Table 4: **Overview of the versions of LLMs evaluated.** For which there are not multiple versions of the model, we have not listed.

C.2 Instructions for analysis in the Sec. 4.2

The instructions for analysis in the Sec. 4.2 are shown in the Figures 16 to 19.

C.3 Instructions for analysis in the Sec. 4.3

This analysis uses the same instructions as in the Sec. 4.2.

C.4 Instructions for analyzing subtle modifications in the Sec. 4.4

The instructions for analyzing subtle modifications in the Sec. 4.4 are shown in the Figure 20

C.5 In-content examples and instructions in the Sec. 5.1

The in-content examples are shown in the Figure 21. The instructions in the Sec. 5.1 are shown in the Figure 22.

C.6 Instructions for multi-instruction tuning in the Sec. 5.2

The instructions for multi-instruction tuning in the Sec. 5.2 are shown in the Figures 23 to 25

Evaluation Prompt

I will give you a given content and a rewritten content. You should judge whether the semantics of these two contents are *strict consistent*. And you must answer me only with the following JSON format:

```
```json
{{
    Thought: Str;
    //The analysis of whether these two contens are consistent.
    Consistency: Boolean;
    //Whether these two contents are consistent.
}}
```
```

Please be aware that the following is the fundamental rules you *must* adhere to when judging:

```
```
1. Do not alter the semantics of the given content.
2. Do not alter the formatting requirements within the content.
3. Do not modify any proper nouns, such as names of people or places.
4. Do not return content that hasn't changed in any way.
5. The contents must all be authentic in expression and logical to the native speaker.

```
```

Content 1:

```
```
{content_1}
```
```

Content 2:

```
```
{content_2}
```
```

Figure 10: **Prompt template for guiding GPT-4 to evaluate the generated content.**

ARC-Challenge

Question: Which technology was developed most recently?
A: cellular telephone
B: television
C: refrigerator
D: airplane
Answer: A

Figure 11: **An example of the ARC-Challenge dataset.**

CommonsenseQA

Question: The sanctions against the school were a punishing blow, and they seemed to what the efforts the school had made to change?
A: ignore
B: enforce
C: authoritarian
D: yell at
E. avoid
Answer: A

Figure 12: **An example of the CommonsenseQA dataset.**

WinoGrande

Assess which sentence is more suitable:
A. Sarah was a much better surgeon than Maria so Sarah always got the harder cases.
B. Sarah was a much better surgeon than Maria so Maria always got the harder cases.
Answer: A

Figure 13: **An example of the WinoGrande dataset.**

GSM8K

Question:
A robe takes 2 bolts of blue fiber and half that much white fiber. How many bolts in total does it take?
Answer
It takes 2/2=«2/2=1»1 bolt of white fiber
So the total amount of fabric is 2+1=«2+1=3»3 bolts of fabric
3

Figure 14: **An example of the GSM8K dataset.**

HumanEval

Below is an instruction that describes a task. Write a response that appropriately completes the request.

```
def how_many_times(string: str, substring: str) -> int:
    """ Find how many times a given substring can be found in the original string. Count
    overlaping cases.
    >>> how_many_times('', 'a')
    0
    >>> how_many_times('aaa', 'a')
    3
    >>> how_many_times('aaaa', 'aa')
    3
    """
```

Response:

```
    times = 0

    for i in range(len(string) - len(substring) + 1):
        if string[i:i+len(substring)] == substring:
            times += 1

    return times
```

Figure 15: **An example of the HumanEval dataset.**

ARC-Challenge

Instruction 1:

```
Answer the following question: {question}
A. {textA}
B. {textB}
C. {textC}
D. {textD}
Answer:
```

Instruction 2:

```
Could you provide a response to the following query: {question} A. {textA} B. {textB} C. {textC}
D. {textD} The response would be:
```

Instruction 3:

```
Regarding this question: {question}
Please provide an answer.
(A) {textA}
(B) {textB}
(C) {textC}
(D) {textD}
Your response:
```

Instruction 4:

```
Question: {question}
A. {textA}
B. {textB}
C. {textC}
D. {textD}
Answer:
```

Instruction 5:

```
{question}
A. {textA}
B. {textB}
C. {textC}
D. {textD}
Answer:
```

Figure 16: **Instructions for the ARC-Challenge dataset.**

WinoGrande

Instruction 1:

```
Identify the superior sentence:
A. {opt1}
B. {opt2}
Response:
```

Instruction 2:

```
Assess which sentence is more suitable:
A. {opt1}
B. {opt2}
Response:
```

Instruction 3:

```
Evaluate which of the following sentences is more appropriate:
A. {opt1}
B. {opt2}
Your answer:
```

Instruction 4:

```
Determine the more appropriate sentence: A. {opt1} B. {opt2}
```

Instruction 5:

```
which of the following sentences is more appropriate:
A. {opt1}
B. {opt2}
```

Figure 17: **Instructions for the WinoGrande dataset.**

GSM8K

Instruction 1:

```
Question: {question}\nLet's think step by step\nAnswer:
```

Instruction 2:

```
I'm going to give you a math problem
Question: {question}
Let's think step by step
Answer:
```

Instruction 3:

```
Please help me to solve this problem
Problem: {question}
Let's think step by step
Response:
```

Instruction 4:

```
{question}
Let's think step by step
Answer:
```

Instruction 5:

```
Q: {question}

Let's think step by step

A:
```

Figure 18: **Instructions for the GSM8K dataset.**

HumanEval

Instruction 1:

```
Below is an instruction that describes a task. Write a response that appropriately completes the
request.

### Instruction:
Create a Python script for this problem:
{prompt}

### Response:
```

Instruction 2:

```
The given task involves creating a Python script to address the specified problem. The instruction
is as follows:

### Instruction:
Create a Python script for this problem:
{prompt}

### Response:
```

Instruction 3:

```
Complete the following python code:
{prompt}
```

Instruction 4:

```
A Python script for this problem should be created based on the following instruction:
{prompt}
```

Instruction 5:

```
Considering the task, generate a Python script based on the provided problem statement:

{prompt}
```

Figure 19: **Instructions for the HumanEval dataset.**

ARC-Challenge

Instruction 1:

```
Below is an instruction that describes a task. Write a response that appropriately completes the
request.

### Instruction:
Create a Python script for this problem:
{prompt}

### Response:
```

Instruction 2:

```
The given task involves creating a Python script to address the specified problem. The instruction
is as follows:

### Instruction:
Create a Python script for this problem:
{prompt}

### Response:
```

Instruction 3:

```
Complete the following python code:
{prompt}
```

Instruction 4:

```
A Python script for this problem should be created based on the following instruction:
{prompt}
```

Instruction 5:

```
Considering the task, generate a Python script based on the provided problem statement:

{prompt}
```

Figure 20: **Instructions with subtle modifications for the ARC-Challenge dataset.**

In-content examples

Example 1: The sanctions against the school were a punishing blow, and they seemed to what the efforts the school had made to change?
A. ignore
B. enforce
C. authoritarian
D. yell at
E. avoid
Answer: A
Example 2: Sammy wanted to go to where the people were. Where might he go?
A. race track
B. populated areas
C. the desert
D. apartment
E. roadblock
Answer: B
Example 3: To locate a choker not located in a jewelry box or boutique where would you go?
A. jewelry store
B. neck
C. jewlery box
D. jewelry box
E. boutique
Answer: A
Example 4: Google Maps and other highway and street GPS services have replaced what?
A. united states
B. mexico
C. countryside
D. atlas
E. oceans
Answer: D
Example 5: The fox walked from the city into the forest, what was it looking for?
A. pretty flowers.
B. hen house
C. natural habitat
D. storybook
E. dense forest
Answer: C

Figure 21: **In-content examples for the Commonsense dataset.** For a given shot x, the $x-shot$ utilizes the first x examples.

CommonsenseQA

Instruction 1:

```
Answer the following question: {question}
A. {A}
B. {B}
C. {C}
D. {D}
E. {E}
Answer:
```

Instruction 2:

```
Could you provide a response to the following query: {question} A. {A} B. {B} C. {C} D. {D} E. {E}
The response would be:
```

Instruction 3:

```
Regarding this question: {question}
Please provide an answer.
(A) {A}
(B) {B}
(C) {C}
(D) {D}
(E) {E}
Your response:
```

Instruction 4:

```
Question: {question}
A. {A}
B. {B}
C. {C}
D. {D}
E. {E}
Answer:
```

Instruction 5:

```
{question}
A. {A}
B. {B}
C. {C}
D. {D}
E. {E}
Answer:
```

Figure 22: **Instructions for the CommonsenseQA dataset.**

TEST-5

Instruction 1:

```
Answer the following question: {question}\nA. {A}\nB. {B}\nC.
{C}\nD. {D}\nE. {E}\nAnswer:
```

Instruction 2:

```
Could you provide a response to the following query: {question}
A. {A} B. {B} C. {C} D. {D} E. {E} The response would be:
```

Instruction 3:

```
Regarding this question: {question}\nPlease provide an
answer.\n(A) {A}\n(B) {B}\n(C) {C}\n(D) {D}\n(E) {E}\nYour
response:
```

Instruction 4:

```
Question: {question}\nA. {A}\nB. {B}\nC. {C}\nD. {D}\nE.
{E}\nAnswer:
```

Instruction 5:

```
{question}\nA. {A}\nB. {B}\nC. {C}\nD. {D}\nE. {E}\nAnswer:
```

Figure 23: **Instructions used in *TEST-5* mode.**

TRAIN-5

Instruction 1:

```
Inquiry: {question}\nA. {A}\nB. {B}\nC. {C}\nD. {D}\nE.
{E}\nResponse:
```

Instruction 2:

```
Please reply to the question that follows:
{question}\n\nChoices:\nA. {A}\nB. {B}\nC. {C}\nD. {D}\nE.
{E}\n\nCorrect Response:
```

Instruction 3:

```
{question}\nA. {A}\nB. {B}\nC. {C}\nD. {D}\nE. {E}\nAnswer:
```

Instruction 4:

```
Please respond to: {question} A) {A} B) {B} C) {C} D) {D} E.
{E} Answer:
```

Instruction 5:

```
Address the following question: {question}\nA. {A}\nB. {B}\nC.
{C}\nD. {D}\nE. {E}\nResponse:
```

Figure 24: **Instructions used in *TRAIN-5* mode.**

TRAIN-10

Instruction 1:

```
Inquiry: {question}\nA. {A}\nB. {B}\nC. {C}\nD. {D}\nE.
{E}\nResponse:
```

Instruction 2:

```
Please reply to the question that follows:
{question}\n\nChoices:\nA. {A}\nB. {B}\nC. {C}\nD. {D}\nE.
{E}\n\nCorrect Response:
```

Instruction 3:

```
{question}\nA. {A}\nB. {B}\nC. {C}\nD. {D}\nE. {E}\nAnswer:
```

Instruction 4:

```
Please respond to: {question} A) {A} B) {B} C) {C} D) {D} E.
{E} Answer:
```

Instruction 5:

```
Address the following question: {question}\nA. {A}\nB. {B}\nC.
{C}\nD. {D}\nE. {E}\nResponse:
```

Instruction 6:

```
Address the following question: {question}\n\nA. {A}\n\nB.
{B}\n\nC. {C}\n\nD. {D}\n\nResponse:
```

Instruction 7:

```
Ponder over the following inquiry: {question}\nA. {A}\nB.
{B}\nC. {C}\nD. {D}\nE. {E}\nAnswer:
```

Instruction 8:

```
Please provide an answer to the question below: {question}\n(A)
{A}\n(B) {B}\n(C) {C}\n(D) {D}\n(E) {E}\nIndicate your answer:
```

Instruction 9:

```
Ponder the following: {question}\nA. {A}\nB. {B}\nC. {C}\nD.
{D}\nE. {E}\nAnswer:
```

Instruction 10:

```
Reply to this inquiry: {question}\nA. {A}\nB. {B}\nC. {C}\nD.
{D}\nE. {E}\nResponse:
```

Figure 25: **Instructions used in *TRAIN-10* mode.**