

RoParQ: Paraphrase-Aware Alignment of Large Language Models Towards Robustness to Paraphrased Questions

Minjoon Choi

Seoul National University
minjoonchoi08@snu.ac.kr

Abstract

Large Language Models (LLMs) often exhibit inconsistent behavior when answering paraphrased questions, suggesting a reliance on surface-level patterns rather than true semantic understanding. To address this limitation, we introduce **RoParQ**, a benchmark specifically constructed to evaluate cross-paraphrase consistency in closed-book multiple-choice QA. This benchmark is derived from standard datasets by generating paraphrases via proprietary models and selectively retaining examples that elicit inconsistent confidence from a judge model. We further propose **XParaCon**, a novel evaluation metric that quantifies a model’s robustness by measuring the standard deviation of accuracies across question variants. Additionally, we implement a reasoning-based, paraphrase-aware Supervised Fine-Tuning (SFT) strategy designed to align models toward semantic invariance. Our experiments demonstrate that this targeted alignment significantly enhances robustness. Notably, fine-tuned lightweight models achieved consistency levels comparable to much larger pre-trained models. These results highlight the efficacy of our approach in mitigating superficial memorization and fostering more robust, reliable LLMs.¹

1 Introduction

Large Language Models (LLMs) have demonstrated remarkable capabilities in natural language understanding and reasoning (Wei et al., 2022), achieving high accuracy across various benchmarks (Brown et al., 2020). However, they still exhibit sensitivity to superficial variations in the given query, which is a critical limitation (Elazar et al., 2021; Sclar et al., 2024). Despite possessing the parametric knowledge required to answer a query correctly, LLMs often exhibit inconsistent behavior when the same question is rephrased, even when

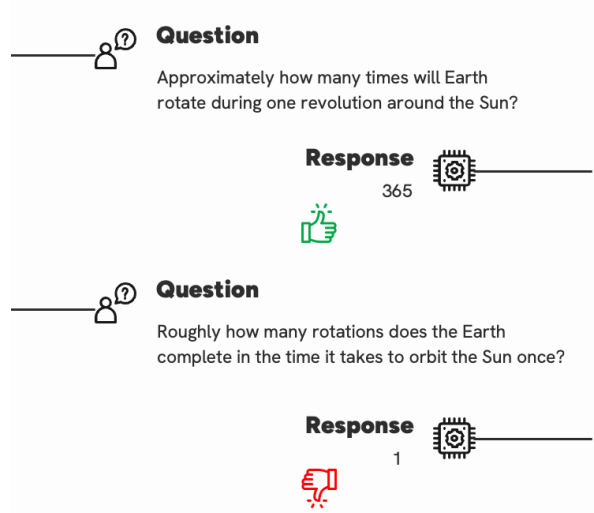


Figure 1: An example of the LLM generating an incorrect response when given a paraphrased question.

the semantic meaning remains identical, as shown in Figure 1. This phenomenon suggests that their high performance on benchmarks may often stem from memorizing superficial answer patterns associated with specific phrasings (McCoy et al., 2024), rather than acquiring true semantic understanding (Berglund et al., 2024). This lack of robustness reveals that models may rely on surface-level cues, raising concerns about their reliability in real-world applications where user queries may be highly diverse. Addressing this inconsistency is essential for developing models that are not only accurate but also robust and semantically invariant.

In this work, we investigate and mitigate this issue by constructing a comprehensive benchmark and proposing a targeted alignment strategy. We introduce **RoParQ** (**R**obustness to **P**araphrased **Q**uestions)², a dataset constructed by paraphrasing questions from established benchmarks (MMLU, ARC, CommonsenseQA, MathQA) using ad-

¹We release our code at: <https://github.com/m-joon-ixix/RoParQ>

²We release our benchmark through Hugging Face: <https://huggingface.co/datasets/m-joon-ixix/RoParQ>

vanced proprietary models. By employing an open-source judge LLM, we specifically filtered for examples that elicited inconsistent confidence, where models answered correctly on some phrasing variants but failed to do so on others. Through this deliberate filtering, we ensure that our dataset targets the precise weakness of current LLMs. To rigorously quantify model stability, we propose a new metric, *XParaCon* (Cross-Paraphrase Consistency), which measures the standard deviation of accuracies across question variants. We first analyze the performance of various pre-trained open-source and proprietary models, observing that while robustness typically scales with parameter count, gaps in consistency across paraphrases still persist. Furthermore, we implement a reasoning-based, paraphrase-aware Supervised Fine-Tuning (SFT) approach on open-source models. Our experimental results demonstrate that this alignment technique significantly improves robustness; for instance, the Llama-3.1-8B-Instruct model’s *XParaCon* score rises from 2.186 to 2.629 by alignment. These results demonstrated our method’s effectiveness in enabling smaller models to achieve consistency levels comparable to, or even exceeding, those of much larger pre-trained models.

The main contributions of this paper are summarized as following:

- (1) **Construction of the RoParQ Benchmark:** We present a novel dataset enriched with high-quality paraphrases generated by proprietary models to rigorously test cross-paraphrase consistency.
- (2) **Proposal of the *XParaCon* Metric:** We introduce a robust evaluation metric that provides a precise and intuitive measure of a model’s semantic invariance.
- (3) **Efficacy of Paraphrase-Aware Alignment:** We demonstrate that our reasoning-based SFT method effectively mitigates inconsistency, showing that explicitly training models on paraphrase invariance enables lightweight models to achieve robust performance comparable to that of larger models.

2 Task Formulation

Given the set of questions $q_i \in \mathbb{Q}$ in the dataset, we denote each question as $q_{original}$. Each $q_{original}$ is paraphrased by two proprietary models; Gemini 2.5 Flash Lite (Comanici et al., 2025) and Claude 3.5 Sonnet (Anthropic, 2024); resulting in q_{gemini} and q_{claude} , respectively. Eventually, a set of questions

$\{q_1, q_2, q_3\}$ is formed where $q_1 = q_{original}$, $q_2 = q_{gemini}$, and $q_3 = q_{claude}$.

Since the task on which this study focuses is multiple-choice question answering, each data example consists of multiple choices $\mathbb{C} = \{c_1, c_2, \dots, c_k\}$. To remove the effect of the ordering of choices given in the prompt, we construct 8 randomly shuffled permutations of choices: $\mathbb{C}_1, \mathbb{C}_2, \dots, \mathbb{C}_8$. The prompt given to the LLM (f) consists of a question q and a permutation of choices \mathbb{C}_i , so the generated response by the LLM can be denoted as $f(q, \mathbb{C}_i)$.

We define the LLM to be “*perfectly correct*” on question q , if the LLM outputs the correct answer when given any of the 8 permutations of choices. This definition can be written as Equation 1.

$$\forall i \in \{1, 2, \dots, 8\}, f(q, \mathbb{C}_i) = c_{GroundTruth} \quad (1)$$

$$\iff \text{“}f \text{ is perfectly correct on question } q\text{”}$$

Using Llama-3.1-8B-Instruct (Grattafiori et al., 2024) as the open-source “judge” LLM, we determine whether the questions $\{q_1, q_2, q_3\}$ of a certain data example are perfectly correct. If all three are perfectly correct, the LLM is considered to be “consistently confident” on this example. If none are perfectly correct, the case is that the model is “consistently unconfident”. These two cases are not in the scope of interest of this study. If one or two of the questions are perfectly correct, we interpret that this example yields “*inconsistent confidence*” from the LLM. Since this is the main target for investigation in our study, these examples are included in our final dataset.

3 Experimental Setup

3.1 Datasets

The questions and multiple-choice options used to form our dataset was sourced from Unified MCQA³. Among the subsets where each one represents the number of choices forming the question, we use the *4-choice* and *5-choice* subsets to ensure a sufficient number of options. Data examples where the source dataset is MMLU (Hendrycks et al., 2021) or ARC (Clark et al., 2018) are employed from the *4-choice* subset. Examples sourced from CommonsenseQA (Talmor et al., 2019) or MathQA (Amini et al., 2019) are selected to be used from the *5-choice* subset. MMLU, ARC, and CommonsenseQA are benchmarks that query

³<https://huggingface.co/datasets/pszemraj/unified-mcqa>

	small-sized	medium-sized	large-sized
Llama 3.1	Llama-3.1-8B-Instruct	Llama-3.1-70B-Instruct	Llama-3.1-405B
Qwen 3	Qwen3-4B-Instruct-2507	Qwen3-30B-A3B-Instruct-2507	-
Mistral	Mistral-7B-Instruct-v0.3	Mistral-Small-24B-Instruct-2501	-

Table 1: List of open-source LLMs used in our study.

for commonsense, real-world general knowledge. MathQA requires step by step reasoning involving elementary calculation in order to reach the final answer.

3.2 Models

In this study, we take multiple open-source, autoregressive transformer-based LLMs into consideration. Specifically, Llama 3.1 (Grattafiori et al., 2024), Qwen 3 (Yang et al., 2025), and Mistral (Mistral AI Team, 2023, 2025) model families are selected for analysis. In each model family, a small-sized model which has less than 10 billion parameters and a medium-sized model which has a parameter size larger than 20 billion is in the scope of our analysis, for the purpose of comparison with respect to model sizes. For the Llama 3.1 family, Llama-3.1-405B was also used in our experiments. The open-source LLMs used in our study are listed in Table 1. In order to compare the performance regarding robustness to paraphrased questions, we expand our experiments to Deepseek-R1 (DeepSeek-AI et al., 2025) and two proprietary models: Gemini 2.5 Flash Lite and Claude 3.5 Sonnet⁴.

3.3 Evaluation

To evaluate the correctness, or accuracy of responses generated by the LLMs, we extract the choice that was selected by the model and compare it with the ground truth label. The LLM is instructed to output its final answer after a predefined prefix⁵, in order to eliminate the possibility of errors when parsing generated responses. The overall accuracy of responses is computed as: (Number of correct responses) \div (Number of total responses).

The cross-paraphrase consistency, or robustness is measured using the standard deviation of the ac-

curacies of the three question variants. Given each data example, we compute the accuracy for each of the three paraphrased variants of the question. By taking the standard deviation of these accuracy measures, we aim to indicate how inconsistent the LLM is in answering an identical question represented in multiple paraphrased versions. Since we obtain a standard deviation for each example in a dataset, the average of these standard deviations manifest the overall cross-paraphrase inconsistency of the whole dataset. In order to make our numeric measure truly represent the consistency itself and make the scale more intuitive, we take the log (base 2) of the average value explained above, and change its sign. As a result, we propose *XParaCon* (Cross-Paraphrase Consistency):

$$XParaCon = -\log_2\left(\frac{1}{n} \sum_{i=1}^n STD\right)$$

$$STD = StdDev(acc(q_{i,0}), acc(q_{i,1}), acc(q_{i,2}))$$

where n is the total number of data examples, and $acc(q_{i,j})$ is the LLM’s accuracy on question q_j of the i th data example.

3.4 Reasoning-based Alignment

This study employs reasoning-based, paraphrase-aware Supervised Fine-Tuning (SFT) to align models toward semantic invariance across paraphrased questions. During SFT, each training instance includes both the original question and a paraphrased variant, and the instruction explicitly requires the model to restate the question in its own words, produce a meaning-preserving paraphrase, and verify that the same option is predicted under that paraphrase⁶. This design couples answer selection with an explicit reasoning routine that compares semantics across formulations, encouraging the model to ground its choice in invariant meaning rather than surface cues. By making paraphrase awareness a supervised objective, the approach leverages the model’s latent reasoning ability to reconcile

⁴For the following models, responses were generated by accessing the inference endpoint of Snowflake Cortex API: Llama-3.1-405B, Deepseek-R1, Claude 3.5 Sonnet

⁵The prompts that were used to instruct LLMs are presented in Figures 5 and 6.

⁶The prompts are presented in Figures 8 and 9.

	MMLU	ARC	CommonsenseQA	MathQA
Initially Sourced	99841	3358	9741	29007
After Data Preprocessing	3899	2174	6995	18542
After Data Selection	707	344	2083	7140

Table 2: Number of data examples obtained in each step, throughout the dataset construction process.

multiple phrasings before committing to an answer, thereby reducing cross-paraphrase inconsistency. Through this alignment procedure that trains the model to internalize a paraphrase-consistency check as part of its decision process, we aim to improve robustness to superficial wording changes while preserving the closed-book, multiple-choice QA evaluation setting. LoRA (Hu et al., 2022) was used for Supervised Fine-tuning, and the hyperparameter settings are presented in Tables 6 and 7.

4 Dataset Construction: RoParQ

4.1 Data Preprocessing

We enforce the closed-book multiple-choice QA task to ensure that downstream paraphrasing and robustness measurements are faithful and comparable across items. Source questions are drawn from four source datasets, as described in Section 3.1. Only questions without any accompanying passages are retained to maintain a closed-book setting, eliminating items whose correctness might hinge on retrieved contexts rather than models’ parametric knowledge and inherent reasoning abilities. To remove formats incompatible with paraphrasing, all questions containing underbars are excluded since these typically indicate masked-word prediction tasks that are harder to stably paraphrase into variants with same semantic meanings. Each question must end with a question mark to reduce prompt variability, which also improves paraphrase alignment.

Questions longer than three sentences are filtered out to avoid cases where long narrative contexts are mixed in, which risks shifting the task away from closed-book QA and increases ambiguity in paraphrase generation. To ensure that paraphrasing operations are meaningful, questions shorter than ten words are removed because overly short questions tend to be underspecified and sensitive to minor rewording, thus being difficult to paraphrase without altering semantic content. After these structural filters, the retained pool consists

of well-formed questions that enable controlled paraphrase generation, producing a high-integrity dataset that supports reliable measurement of cross-paraphrase stability.

4.2 Question Paraphrasing

Proprietary models are known for their strong abilities to follow instructions, so Gemini 2.5 Flash Lite and Claude 3.5 Sonnet⁷ are used for the task of question paraphrasing. For each question, we employ the following instruction to generate a paraphrased variant: “*Your task is to paraphrase only the question, keeping the meaning exactly the same but using different wording*”. Extra requirements that restrict edits only to the question and prevent any modification of choices were added to the instruction. A constraint to keep the question type (MCQA) unchanged was also augmented to avoid shifts to other tasks. The QA triplet (Question, Choices, Answer) was provided to the LLM in order to ground the paraphrase in the precise decision context. The exact prompt that was used to query LLMs to paraphrase questions is provided in Figure 7.

4.3 Data Selection

To execute the data selection process, we generate responses for each question variant and random-shuffled permutation of choices. As a result, we generate⁸ 24 responses for each data example, using our judge model Llama-3.1-8B-Instruct. Meanwhile, the consistent accuracy statistics of these generated responses⁹ across $q_{original}$, q_{gemini} , and q_{claude} depict the above par quality of the question paraphrasings. As described in Section 2, examples in which the judge LLM was “*perfectly correct*” on one or two question variants, which is defined as examples yielding “*inconsistent confidence*”, are added to our final dataset, **RoParQ** (Robustness to

⁷Two examples from CommonsenseQA were removed since the Claude model refused to paraphrase due to harmful (porn-related) content residing in those examples.

⁸Greedy decoding (temperature = 0) was used to generate responses.

⁹Accuracy statistics are presented in Table 8.

Model	General Knowledge		Math Reasoning	
	Accuracy	XParaCon	Accuracy	XParaCon
Llama-3.1-8B-Instruct	0.781	2.186	0.738	1.924
Llama-3.1-70B-Instruct	0.869	3.219	0.863	3.114
Llama-3.1-405B	0.882	3.272	0.934	3.762
Mistral-7B-Instruct-v0.3	0.697	2.663	0.344	2.728
Mistral-Small-24B-Instruct-2501	0.855	3.121	0.899	3.435
Qwen3-4B-Instruct-2507	0.802	2.848	<u>0.942</u>	4.489
Qwen3-30B-A3B-Instruct-2507	0.866	3.136	0.935	4.034
Deepseek-R1	0.850	<u>3.378</u>	0.915	4.405
Claude 3.5 Sonnet	<u>0.876</u>	3.428	0.959	5.164
Gemini 2.5 Flash Lite	0.861	3.339	0.941	<u>4.195</u>

Table 3: Performance of pre-trained open-source models and proprietary models on each subset of our benchmark RoParQ. The highest value in each column is highlighted in **bold**, and the second-highest value is underlined.

Paraphrased Questions). The number of examples obtained in each step; initially sourced, after data preprocessing, and after data selection; is shown in Table 2.

Since each source dataset exhibits different characteristics (Section 3.1), we categorize RoParQ into two subsets: *general knowledge* and *math reasoning*. Examples with questions asking for general knowledge or commonsense (MMLU, ARC, and CommonsenseQA) reside in the *general knowledge* subset. On the other hand, examples with queries that require deeper, multi-step numerical reasoning (MathQA) are labeled as the *math reasoning* subset. Furthermore, our benchmark RoParQ consists of three splits: train (70%), validation (15%), and test (15%). The number of examples in each subset and split is presented in Table 4.

	Train	Validation	Test
General Knowledge	2194	470	470
Math Reasoning	4998	1071	1071

Table 4: Number of data examples in each subset/split setting of the RoParQ benchmark.

5 Results

5.1 Baselines

We first present the overall accuracy of responses and *XParaCon* metrics on pre-trained open-source models and proprietary models, in Table 3.

The performance analysis across the two RoParQ subsets revealed distinct patterns. On the *general knowledge* subset, the largest open-source model, Llama-3.1-405B, achieved the high-

est accuracy, followed by Claude 3.5 Sonnet. However, Claude 3.5 Sonnet demonstrated the strongest cross-paraphrase consistency, closely followed by Gemini 2.5 Flash Lite, suggesting superior robustness among the proprietary models in this domain. The trend of scaling performance is especially evident within the Llama 3.1 family, where the 70B model significantly outperforms the 8B model in both accuracy and *XParaCon*, and the 405B model further improves upon both metrics, reinforcing the general principle that performance scales with model size.

Conversely, for the *math reasoning* subset, Claude 3.5 Sonnet was the top performer in both metrics, achieving the highest accuracy and a considerably higher *XParaCon*, with Gemini 2.5 Flash Lite holding the second rank for robustness. These results collectively indicate that the proprietary models, particularly Claude 3.5 Sonnet, generally offer the most consistent and robust performance against paraphrasing, especially in challenging math reasoning tasks, although the largest open-source models remain highly competitive in general knowledge accuracy. A notable exception to the overall hierarchy is the Qwen3-4B-Instruct-2507 model. Despite its relatively small parameter count, this model achieved the second-highest accuracy among all models and a strong *XParaCon* score, suggesting that specialized training techniques, such as strong-to-weak distillation used in the Qwen family, can lead to unexpectedly efficient and effective reasoning capabilities in resource-constrained models for specific tasks.

Model	General Knowledge		Math Reasoning	
	Accuracy	XParaCon	Accuracy	XParaCon
Llama-3.1-8B-Instruct	0.781	2.186	0.738	1.924
Llama-3.1-8B-Instruct (FT)	0.798	2.629	0.685	2.316
Mistral-7B-Instruct-v0.3	0.697	2.663	0.344	2.728
Mistral-7B-Instruct-v0.3 (FT)	0.735	2.854	0.405	2.617
Qwen3-4B-Instruct-2507	0.802	2.848	0.942	4.489
Qwen3-4B-Instruct-2507 (FT)	0.821	2.920	0.951	4.856

Table 5: Performance of open-source models (pre-trained and fine-tuned) on each subset of our benchmark RoParQ. Fine-tuned models are denoted as ‘**FT**’. *XParaCon* scores of fine-tuned models are highlighted in **bold**.

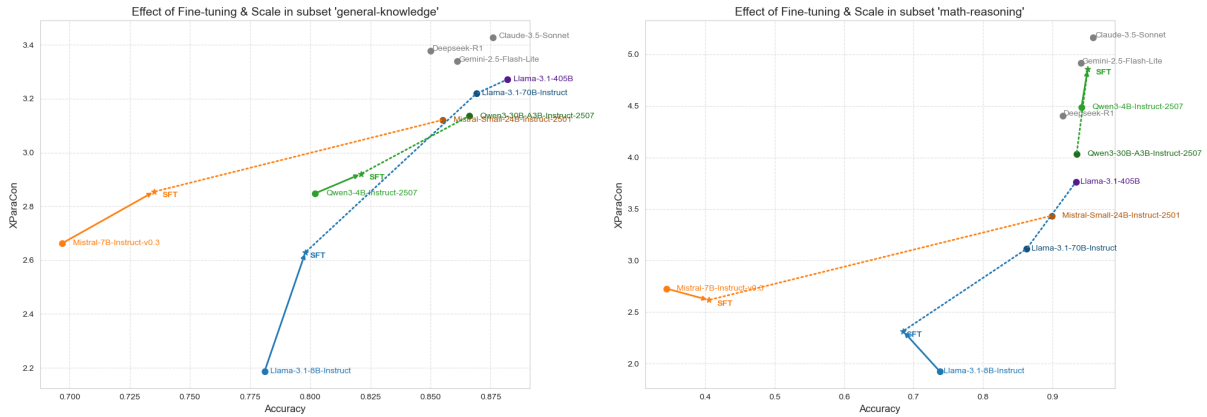


Figure 2: Movements of Accuracies and *XParaCon* scores within each model family, representing the effect of scale and fine-tuning.

5.2 Post-Alignment

Table 5 describes the overall accuracy and *XParaCon* of small-sized open-source models and their fine-tuned checkpoints.¹⁰

The results across both subsets confirm the efficacy of our fine-tuning based approach in improving model robustness to paraphrased questions. On the *general knowledge* subset, fine-tuning yielded a consistent and substantial increase in *XParaCon* scores for all models (Figure 2), often bringing the smaller fine-tuned models’ scores near those of much larger pre-trained models in the same family, effectively leveraging targeted training to compensate for a smaller parameter count. A similar trend was observed in the *math reasoning* subset, where the Qwen3-4B model, for example, saw its *XParaCon* score improve from 4.489 to 4.856. This fine-tuned robustness, particularly for Qwen3-4B, reached a level comparable to, or exceeding, several much larger open-source models, underscoring the power of goal-oriented supervised fine-tuning.

¹⁰Figures 3 and 4 provide a comprehensive view of the *XParaCon* scores for each model.

However, it is noteworthy that for the Llama-3.1-8B-Instruct model on the *math reasoning* subset, this significant boost in *XParaCon* was accompanied by a minor decline in raw accuracy, lighting the possibility of a potential trade-off between consistency and correctness in domain-specific tasks for light-weighted models.

6 Related Work

Alting von Geusau and Bloem (2021) evaluate QA robustness to paraphrased questions, finding consistent performance drops and showing BERT’s sentence embeddings better capture paraphrase similarity than alternatives. Unlike their focus on extractive QA and broad paraphrase coverage, our work targets closed-book multiple-choice QA and focuses on data examples exhibiting cross-paraphrase inconsistency, enabling fine-grained analysis. Gan and Ng (2019) create two SQuAD-based paraphrase datasets and show significant degradation across state-of-the-art models. They also explore data augmentation via neural paraphrasing to improve robustness. In contrast, this work selects

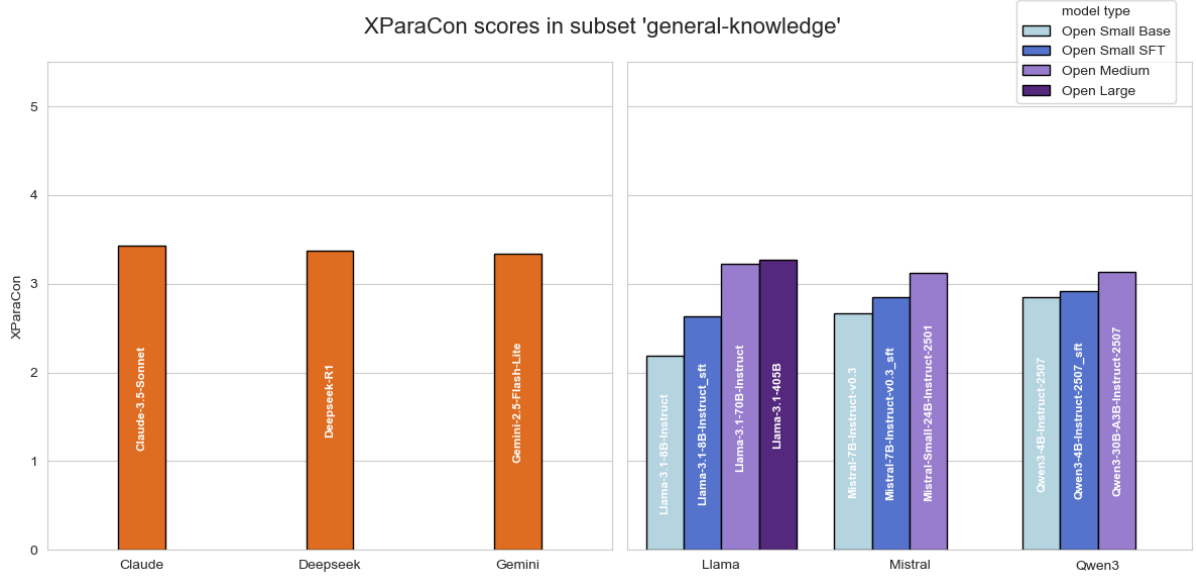


Figure 3: *XParaCon* score of each model in the *general knowledge* subset.

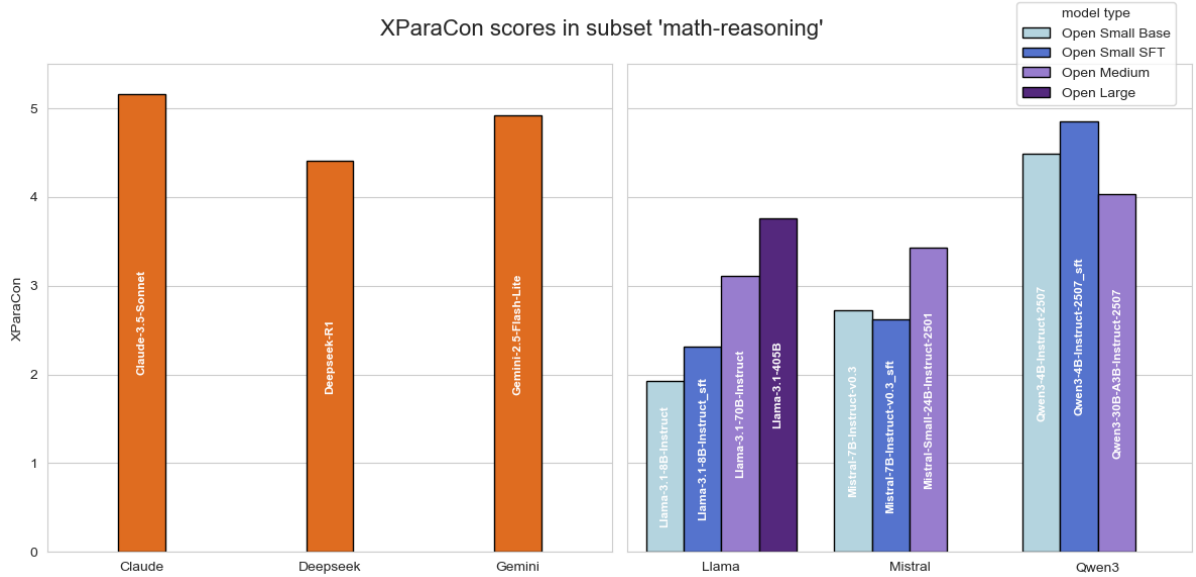


Figure 4: *XParaCon* score of each model in the *math reasoning* subset.

only questions that empirically trigger inconsistency across paraphrases, and further proposes a paraphrase-aware supervised fine-tuning protocol aimed at enforcing answer invariance. [Deng et al. \(2024\)](#) propose Rephrase-and-Respond (RaR), where one LLM rephrases prompts and another generates the answer, improving reliability across diverse tasks. While RaR uses self-rephrasing as an inference-time strategy to enhance performance, our work builds a benchmark to evaluate robustness to paraphrases and introduces a training-time method to align model behavior across paraphrases.

[Dong et al. \(2017\)](#) present a framework that generates candidate question paraphrases from multiple sources and learns a neural scoring model to weight paraphrases in order to improve QA performance. In contrast, our work proposes a paraphrase-aware supervised fine-tuning protocol to enforce answer invariance across question variants, rather than learning to score paraphrases within a retrieval or span-extraction pipeline.

7 Conclusion

This study was motivated by the limitations of Large Language Models regarding their sensitivity to superficial input variations. To address these limitations, we constructed the *RoParQ* benchmark to rigorously test cross-paraphrase consistency and introduced the *XParaCon* metric to quantify it. Furthermore, we proposed a reasoning-based Supervised Fine-Tuning strategy aimed at aligning models toward semantic invariance. Our analysis initially confirmed that robustness generally scales with parameter count. Moreover, our experiments validate the efficacy of our alignment approach, showing that fine-tuning significantly boosts consistency scores. Notably, a fine-tuned model achieved robustness comparable to that of much larger baselines, demonstrating that targeted alignment can effectively liberate semantic stability from the constraints of parameter scale. This establishes a promising direction for developing efficient, reliable LLMs that do not rely on massive scale alone for robustness. Moving forward, this work serves as a foundation for further research into paraphrase-aware training, ultimately steering the field toward models that possess a genuine, generalized understanding of language beyond surface-level patterns.

Limitations

Our study has limitations that guide future research. Firstly, we focused primarily on commonly accessible LLM parameter sizes, reserving the evaluation of significantly larger open-source models for future work, which is necessary to fully assess the potential impact of scalability on robustness. Secondly, while effective, the RoParQ benchmark is limited to multiple-choice questions. Extending it to open-ended questions would broaden its scope. Furthermore, the evaluation was conducted solely in English, necessitating future work to explore robustness in a multilingual context. Our study relied on Supervised Fine-Tuning (SFT). While SFT was successful at dramatically improving consistency scores, future investigations may explore more advanced post-training methods, such as Reinforcement Learning from Human Feedback (RLHF) or other alignment techniques.

Acknowledgments

This work was supported by the Korea Institute for Advancement of Technology (KIAT) grant funded

by the Korean Government (Ministry of Education). (P0025681-G02P22450002201-10054408, Semiconductor-Specialized University)

References

- Paulo Alting von Geusau and Peter Bloem. 2021. Evaluating the robustness of question-answering models to paraphrased questions. In *Artificial Intelligence and Machine Learning*, pages 1–14, Cham. Springer International Publishing.
- Aida Amini, Saadia Gabriel, Shanchuan Lin, Rik Koncel-Kedziorski, Yejin Choi, and Hannaneh Hajishirzi. 2019. *MathQA: Towards interpretable math word problem solving with operation-based formalisms*. 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)*, pages 2357–2367, Minneapolis, Minnesota. Association for Computational Linguistics.
- Anthropic. 2024. Claude 3.5 sonnet. <https://www.anthropic.com/news/claude-3-5-sonnet>. Technical Release Document/Model Card Addendum, Accessed: 2025-11-25.
- Lukas Berglund, Meg Tong, Maximilian Kaufmann, Mikita Balesni, Asa Cooper Stickland, Tomasz Korbak, and Owain Evans. 2024. *The reversal curse: LLMs trained on “a is b” fail to learn “b is a”*. In *The Twelfth International Conference on Learning Representations*.
- Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D 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 Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. *Language models are few-shot learners*. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.
- 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*. Preprint, arXiv:1803.05457.
- Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, Luke Marris, Sam Petulla, Colin Gaffney, Asaf Aharoni, Nathan Lintz, Tiago Cardal Pais, Henrik Jacobsson, Idan Szpektor, Nan-Jiang Jiang, and 3416 others. 2025. *Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities*. Preprint, arXiv:2507.06261.

- DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, and 181 others. 2025. [Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning](#). *Preprint*, arXiv:2501.12948.
- Yihe Deng, Weitong Zhang, Zixiang Chen, and Quanquan Gu. 2024. [Rephrase and respond: Let large language models ask better questions for themselves](#). *Preprint*, arXiv:2311.04205.
- Li Dong, Jonathan Mallinson, Siva Reddy, and Mirella Lapata. 2017. [Learning to paraphrase for question answering](#). In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 875–886, Copenhagen, Denmark. Association for Computational Linguistics.
- Yanai Elazar, Nora Kassner, Shauli Ravfogel, Abhishek Ravichander, Eduard Hovy, Hinrich Schütze, and Yoav Goldberg. 2021. [Measuring and improving consistency in pretrained language models](#). *Transactions of the Association for Computational Linguistics*, 9:1012–1031.
- Wee Chung Gan and Hwee Tou Ng. 2019. [Improving the robustness of question answering systems to question paraphrasing](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6065–6075, Florence, Italy. Association for Computational Linguistics.
- Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. [The llama 3 herd of models](#). *Preprint*, arXiv:2407.21783.
- Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. [Measuring massive multitask language understanding](#). In *International Conference on Learning Representations*.
- Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. [LoRA: Low-rank adaptation of large language models](#). In *International Conference on Learning Representations*.
- R. Thomas McCoy, Shunyu Yao, Dan Friedman, Mathew D. Hardy, and Thomas L. Griffiths. 2024. [Embers of autoregression show how large language models are shaped by the problem they are trained to solve](#). *Proceedings of the National Academy of Sciences*, 121(41):e2322420121.
- Mistral AI Team. 2023. Introducing mistral 7b. <https://mistral.ai/news/announcing-mistral-7b>. Blog post, Accessed: 2025-11-25.
- Mistral AI Team. 2025. Introducing mistral small 3. <https://mistral.ai/news/mistral-small-3>. Blog post, Accessed: 2025-11-25.
- Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. 2024. [Quantifying language models’ sensitivity to spurious features in prompt design or: How i learned to start worrying about prompt formatting](#). In *The Twelfth International Conference on Learning Representations*.
- Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. [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)*, pages 4149–4158, Minneapolis, Minnesota. Association for Computational Linguistics.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed H. Chi, Quoc V Le, and Denny Zhou. 2022. [Chain of thought prompting elicits reasoning in large language models](#). In *Advances in Neural Information Processing Systems*.
- An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. [Qwen3 technical report](#). *Preprint*, arXiv:2505.09388.

A Appendix

Hyperparameter	Value
seed	42
learning_rate	0.0002
lr_scheduler_type	linear
per_device_train_batch_size	1
per_device_eval_batch_size	1
warmup_ratio	0.03
weight_decay	0
logging_strategy	steps
logging_steps	100
eval_strategy	steps
eval_steps	500
save_strategy	steps
save_steps	500
save_total_limit	2
load_best_model_at_end	TRUE
metric_for_best_model	eval_loss
greater_is_better	FALSE
gradient_accumulation_steps	1
max_grad_norm	1
bf16	TRUE

Table 6: List of hyperparameters used for Supervised Fine-tuning (SFT).

You are given a multiple-choice question. Select the single best choice and briefly justify it step by step. Keep your reasoning concise and output the answer preceded by the exact prefix 'Answer: '.

Here is the question:

Question: {question}

Choices:
{choices}

Now, generate the answer and reasoning in the exact format:

Answer: [Upper Case Letter]. [Choice Text]

Reasoning: [Concise Sentences]

Figure 5: Prompt used for multiple choice question answering in the *general knowledge* subset.

You are given a multiple-choice question. Select the single best choice and briefly justify it step by step. Keep your reasoning as concise as possible and output the answer preceded by the exact prefix 'Answer: '.

Here is the question:

Question: {question}

Choices:
{choices}

Now, generate the reasoning and answer in the exact format:

Reasoning: [Concise Sentences]

Answer: [Upper Case Letter]. [Choice Text]

Figure 6: Prompt used for multiple choice question answering in the *math reasoning* subset. The model is instructed to generate its reasoning first since questions in this subset necessarily require step by step reasoning.

Hyperparameter	Value
r	16
lora_alpha	32
lora_dropout	0.05
bias	none
task_type	CAUSAL_LM

Table 7: List of LoRA hyperparameters used for Supervised Fine-tuning (SFT).

You are given a multiple-choice question with multiple choices and a correct answer. Your task is to paraphrase only the question, keeping the meaning exactly the same but using different wording. You must not change, reorder, or edit the choices or the answer in any way. Do not add new information. Do not change the question type: if it is a multiple-choice question, it must remain so. Output the paraphrased question preceded by the exact prefix 'New Question: '.

Here is the original content:

Question: {question}

Choices:
{choices}

Answer: {answer}

Now, generate the paraphrased question in the format:
New Question: [Paraphrased question]

Figure 7: Prompt used for question paraphrasing. The model is instructed to output the paraphrased question in a fixed format.

You are given a multiple-choice question. Before selecting an option, restate the question in your own words and list a paraphrase that preserves its meaning; use this to verify that the same choice remains correct under the paraphrase. Select the single best choice and briefly justify it step by step. Keep your reasoning concise and output the answer preceded by the exact prefix 'Answer: '.

Here is the question:

Question: {question}

Choices:
{choices}

Now, generate the paraphrase, answer, and reasoning in the exact format:
Paraphrase: [A rewrite that preserves meaning]
Answer: [Upper Case Letter]. [Choice Text]
Reasoning: [Concise Sentences]

Figure 8: Prompt used for paraphrase-aware multiple choice question answering in the *general knowledge* subset. This prompt was used for fine-tuning and inference on fine-tuned models.

	General Knowledge			Math Reasoning		
	<i>qoriginal</i>	<i>qgemini</i>	<i>qclaude</i>	<i>qoriginal</i>	<i>qgemini</i>	<i>qclaude</i>
Overall Accuracy	0.785	0.770	0.792	0.597	0.538	0.582
Ratio of Perfectly Correct Examples	0.646	0.636	0.664	0.380	0.317	0.369

Table 8: Accuracy statistics on responses generated by Llama-3.1-8B-Instruct to select examples that yield inconsistent confidence.

You are given a multiple-choice question. Before selecting an option, restate the question in your own words and list a paraphrase that preserves its meaning; use this to verify that the same choice remains correct under the paraphrase. Select the single best choice and briefly justify it step by step. Keep your reasoning as concise as possible and output the answer preceded by the exact prefix 'Answer: '.

Here is the question:

Question: {question}

Choices:
{choices}

Now, generate the paraphrase, reasoning, and answer in the exact format:
Paraphrase: [A rewrite that preserves meaning]
Reasoning: [Concise Sentences]
Answer: [Upper Case Letter]. [Choice Text]

Figure 9: Prompt used for paraphrase-aware multiple choice question answering in the *math reasoning* subset. This prompt was used for fine-tuning and inference on fine-tuned models.