Benchmarking Open LLMs for Automated Evaluation: Towards Reliable and Accessible Model Assessment

Anonymous ACL submission

Abstract

Large language models (LLMs) have become prevalent in natural language processing, with researchers increasingly using them as automated evaluators through the LLM-as-a-judge paradigm. However, current implementations primarily rely on proprietary models, raising concerns about accessibility, costs, and data privacy. Additionally, existing LLM judges exhibit various biases that can compromise evaluation quality. We systematically investigate whether general-purpose open LLMs, without specific fine-tuning for evaluation tasks, can serve as reliable alternatives to proprietary models. We conduct comprehensive assessments across established benchmarks and analyze their susceptibility to different biases. Our findings demonstrate that certain open models can match or exceed the performance of proprietary alternatives, providing a systematic methodology for selecting appropriate open-source evaluators while maintaining high standards of assessment quality.

1 Introduction

Large language models (LLMs) have revolutionized natural language processing with their broad capabilities, but evaluating their performance presents significant challenges. Unlike traditional machine learning models with clear metrics, LLMs require more nuanced evaluation approaches that often rely on human feedback. However, human evaluation introduces substantial drawbacks: it is expensive, time-consuming, and difficult to scale.

To address these limitations, researchers have proposed using LLMs themselves as evaluators, known as the *LLM-as-a-judge* paradigm (Zheng et al., 2023). While this approach shows promise in automating evaluations and maintaining alignment with human judgment (Chang et al., 2024), it faces several critical challenges. Current implementations primarily rely on proprietary models, raising concerns about accessibility, costs, and data privacy. Additionally, existing LLM judges exhibit various biases, including sensitivity to response length (Dubois et al., 2024), order effects (Zheng et al., 2023), and preferences (Liu et al., 2023).

In this work, we systematically investigate whether general-purpose open LLMs, without specific fine-tuning for evaluation tasks, can serve as reliable alternatives to proprietary models. We conduct comprehensive assessments of leading open models across established benchmarks, examining their effectiveness as judges and analyzing their susceptibility to various biases. While some open models claim comparable performance to proprietary alternatives (Thakur et al., 2024), our study provides rigorous validation of these claims in an evaluation context.

Our findings have important implications for the broader adoption of automated evaluation systems. Through rigorous assessment of open LLM judges across multiple dimensions of reliability, we argue in favor of their viability as cost-effective alternatives to proprietary models. This work also provides a method for selecting appropriate evaluators, establishing a foundation for broader adoption of open models in evaluation tasks.

2 LLM-as-a-Judge

Recent advancements in large language models, such as GPT-4 (Achiam et al., 2023), have demonstrated remarkable improvements in instruction following, query understanding, and response generation (Li et al., 2025). While these capabilities make evaluating such models increasingly challenging, they have simultaneously created an opportunity: researchers have begun leveraging these advanced LLMs as scalable automated evaluators, a paradigm known as *LLM-as-a-judge* (Zheng et al., 2023).

This approach has proven particularly promising due to modern LLMs' strong alignment with hu-

man preferences, achieved through reinforcement learning from human feedback (RLHF) (Murugadoss et al., 2024). As a result, LLM judges show high correlation with human judgments even with minimal evaluation instructions (Li et al., 2025).

Beyond evaluation, LLM-based judges play a crucial role in model alignment, retrieval, and reasoning tasks (Bai et al., 2022; Li and Qiu, 2023; Liang et al., 2023; Lee et al., 2024; Li et al., 2024; Zhao et al., 2024), offering a cost-effective solution for comparing outputs across models. In model alignment, LLM-based evaluators help guide fine-tuning processes by identifying discrepancies between generated responses and human expectations. In retrieval tasks, they assess the relevance and quality of retrieved information, ranking results to improve search quality. For reasoning tasks, LLM judges validate logical consistency and correctness, ensuring that models produce factually accurate and well-structured outputs.

LLM-as-a-judge frameworks employ two main evaluation methodologies to ensure systematic and consistent assessment: absolute evaluation and comparative evaluation (Li et al., 2025). In absolute evaluation, responses are graded individually against predefined criteria. However, this approach often struggles to capture nuanced differences between responses and tends to produce unstable results, as scores can vary significantly when using different judge models (Zheng et al., 2023). Comparative evaluation, which has become the predominant approach, involves directly comparing two or more responses to determine their relative quality. Formally, given a judge LLM J, the comparative assessment process can be expressed as:

$$R = J(C_1, \ldots, C_n), \tag{1}$$

where C_i represents the ith candidate response being evaluated, and R denotes the evaluation result. The process can take two forms: pairwise evaluation ($n = 2$), where two responses are compared directly, or list-wise evaluation ($n > 2$), where multiple responses are ranked simultaneously (Zheng et al., 2023; Shen et al., 2024).

3 Can You Trust LLM Judgments?

The reliability of LLM-based judges can be systematically evaluated through several dimensions that collectively determine their trustworthiness: position bias assessment measures consistency across different response orderings, instruction following capabilities verify accurate interpretation and application of evaluation criteria while resisting superficial features, performance on challenging tasks demonstrates sophisticated reasoning abilities, and human alignment ensures judgments match expert evaluations. Together, these comprehensive assessments validate whether an LLM can serve as a dependable evaluator across diverse applications.

Position bias. A critical challenge in LLM evaluation is position bias, where the ordering of responses influences the model's judgment independently of response quality. An ideal LLM judge should evaluate responses based solely on their merit, maintaining consistent assessments regardless of presentation order in pairwise comparisons.

To detect position bias, researchers employ double-blind evaluation protocols where identical response pairs are presented in different orders (Zheng et al., 2023). A reliable judge should demonstrate consistent judgments across these permutations. Significant variations in assessments based on ordering indicate susceptibility to positional effects, which can compromise evaluation fairness and reliability.

Instruction following. The ability to accurately interpret and apply evaluation criteria is fundamental to a trustworthy LLM judge. Unlike traditional metrics that rely on fixed scoring rules, LLM judges must process and adhere to diverse, often complex instructions while maintaining evaluation consistency. Poor instruction following can lead to misaligned assessments that fail to capture the intended evaluation criteria.

A particular challenge lies in distinguishing between responses that satisfy task requirements and those that merely demonstrate surface-level competence. For instance, a judge might incorrectly favor stylistically polished but substantively inadequate responses. Therefore, evaluating an LLM judge's instruction-following capabilities is crucial for ensuring that assessments reflect meaningful quality differences rather than superficial features.

Performance in challenging tasks. A robust LLM judge must excel at evaluating responses in scenarios that require sophisticated reasoning and nuanced understanding. Many real-world applications involve ambiguous or multifaceted problems where simple right-or-wrong determinations are insufficient. Judges must demonstrate the ability to assess subtle quality gradations, identify logical

inconsistencies, and maintain evaluation standards across diverse contexts.

The performance on challenging tasks serves as an indicator of an LLM judge's capability to handle edge cases and complex scenarios. Without strong performance in these areas, a judge may default to oversimplified assessments or fail to capture important qualitative differences between responses.

4 Evaluating Open Models

We evaluate open LLMs across three dimensions to assess their alignment with human expectations:

- **Position Bias**: Using the Position Bias Analyzer (Shi et al., 2024), we measure whether LLM judges exhibit bias based on response position rather than content quality. The evaluation uses datasets from MT-Bench (Zheng et al., 2023), with performance measured by the percentage of cases where the model's preference remains consistent when prompt order is changed.

- **Instruction Following**: We apply the LLMBar benchmark (Zeng et al., 2024) containing 419 output pairs to assess how well LLMs follow instructions. The benchmark comprises two components: the Natural Set for real-world scenarios, and the Adversarial Set, designed to challenge evaluators with responses that deviate from instructions while appearing superficially strong. Performance is measured through alignment with human annotations.

- **Complex Reasoning**: Using JudgeBench (Tan et al., 2024), we evaluate LLMs' ability to assess responses requiring advanced reasoning and factual accuracy across knowledge, reasoning, mathematics, and coding domains. The benchmark includes carefully crafted response pairs containing subtle logical errors to test discrimination capabilities. Performance is measured by the accuracy in identifying logically valid responses.

For instruction following and challenging task evaluations, we employ the swapping operation technique (Zheng et al., 2023) to control positional bias. This involves evaluating each response pair twice with reversed ordering and averaging the results. When a model's preference changes after swapping positions, we consider this a "tie" indicating uncertainty in the model's ability to differentiate between responses. This approach helps isolating the model's true evaluation capabilities from any potential position-based preferences.

For each benchmark, we utilized the prompting strategies proposed in their respective works. In the instruction following evaluation, we employed the Metrics+Reference+Rules prompt from Zeng et al. (2024), which demonstrated superior performance in their analysis by establishing explicit evaluation rules, self-defined quality metrics, and reference solution generation to improve their assessment capabilities. For the challenging task assessment, we implemented the arena-hard judge format from Tan et al. (2024), which directs the model to first generate its own reference solution before analyzing candidate responses.

The evaluation focuses on assessing the inherent capabilities of general-purpose models rather than fine-tuned models specifically trained as evaluators. The assessed models include leading families from official providers. These models were accessed via TogetherAPI[1] due to hardware constraints. Selection was based on the top-performing models listed on the Open LLM Leaderboard (Fourrier et al., 2024), ensuring that only officially released versions from their respective providers were considered. All benchmarks were evaluated using their original implementations, sourced from the respective repositories. [2] [3] [4]

Table 1 presents a summary of the evaluation results, with additional details and results provided in Appendix A. The results demonstrate a substantial performance gap between large and small models across all evaluation criteria. Large models (>27B parameters) consistently outperform their smaller counterparts, with the disparity being particularly pronounced in challenging tasks where large models achieve scores ranging from 40.3% to 56.6%, while smaller models generally perform below 40%, with some scoring as low as 3.14%. Notably, Llama3.3:70B emerges as the strongest performer among open models, matching or exceeding GPT-4's performance across multiple metrics, including position bias resistance (88.8% vs 81.5%) and instruction following in both natural (95.5%) and adversarial (83.3%) settings.

Our analysis reveals a strong correlation between performance on adversarial instruction fol-

[1] https://api.together.xyz/

[2] https://github.com/Slimshilin/Position-Bias-Analyzer/tree/main

[3] https://github.com/princeton-nlp/LLMBar

[4] https://github.com/ScalerLab/JudgeBench

	Position Bias	Instruction Following		Challenging Task
		Natural	Adversarial	
GPT-4* / GPT-4o**	81.5%*	96.0%*	83.3%*	56.6%**
Llama3:70B	83.8%	90.0%	76.8%	46.9%
Llama3.1:70B	83.1%	**95.5%**	82.6%	52.3%
Llama3.3:70B	**88.8%**	**95.5%**	**83.3%**	**56.6%**
Gemma2:27B	71.9%	88.0%	75.2%	41.7%
Qwen2:72B	79.4%	93.5%	73.5%	40.3%
Qwen2.5:72B	84.4%	94.5%	73.5%	55.1%
Mixtral:8x7B	76.2%	82.5%	58.5%	46.6%
Llama3:8B	62.5%	81.0%	56.0%	40.3%
Llama3.1:8B	**73.1%**	83.0%	55.2%	40.9%
Gemma2:9B	71.2%	**90.0%**	74.5%	**42.9%**
Qwen2:7B	39.9%	83.5%	54.8%	26.3%
Qwen2.5:7B	38.5%	89.5%	**65.9%**	41.7%
Mistral:7B	58.1%	68.5%	47.4%	17.1%
Granite-3.1:8B	40.5%	79.5%	56.5%	12.3%
Phi-3-small	56.9%	89.5%	61.3%	38.0%
Falcon3:7B	53.5%	83.5%	51.8%	37.1%
Internlm2.5:7b	57.1%	82.0%	57.3%	3.14%

Table 1: **Results of various language models on benchmark tasks**. Results demonstrate that larger models (>27B parameters) consistently outperform smaller models across all evaluation criteria, with Llama3.3:70B matching or exceeding GPT-4's performance in several metrics. At the top, we present large models, while smaller models are listed at the bottom. The best results in each category (large and small models) are highlighted in bold. *For position bias and instruction following we respectively source GPT-4-0613 results from Shi et al. (2024) and Zeng et al. (2024), and **for the challenging task we source GPT-4o results from Tan et al. (2024).

lowing and complex reasoning tasks. Models that excel at maintaining robust instruction adherence under adversarial conditions tend to perform better on complex reasoning tasks. This relationship is exemplified by Llama3.3:70B and GPT-4, which achieve the highest scores in both adversarial instruction following (83.3%) and complex reasoning (56.6%). The particularly low performance of smaller models on complex reasoning tasks (e.g., Mistral:7B at 17.1%, Granite-3.1:8B at 12.3%, and Internlm2.5:7b at 3.14%) may be partially attributed to the evaluation methodology itself. Since the JudgeBench protocol requires models to first generate their own reference solution before evaluating responses, models that struggle with the underlying tasks may produce poor reference solutions, further compromising their ability to make accurate judgments. However, even the best-performing models achieve only moderate success rates on complex reasoning tasks, indicating significant room for improvement in achieving highly reliable complex evaluations.

5 Conclusion

In this study, we evaluate the performance of leading open language models on benchmark tasks, focusing on position bias, instruction following, and performance on challenging tasks. Our results indicate that larger models consistently outperform smaller ones, particularly in complex reasoning tasks. Notably, Llama3.3:70B demonstrated superior performance, matching or exceeding GPT-4 in several metrics, establishing itself as a viable open alternative to the widely used but closed-source GPT-4. This is particularly significant for academic labs that can now run state-of-the-art models locally for research purposes. The correlation between adversarial instruction following and challenging task performance suggests that robust instruction adherence is crucial for complex evaluations. However, even the best models show room for improvement across all dimensions.

Ethical Implications. The superior performance of larger models raises important ethical considerations regarding computational resource allocation and environmental impact. The significant computational requirements for training and running these models could exacerbate existing inequalities in access to AI technologies. Additionally, as these models become more capable of complex reasoning and evaluation tasks, their potential influence on decision-making processes across various domains increases, necessitating careful consideration of fairness, accountability, and transparency in their deployment.

Limitations. Our evaluation criteria relate to other generally desirable LLM properties, such as helpfulness. This paper emphasizes helpfulness but largely neglects safety. Honesty and harmlessness are crucial for a chat assistant as well. Additionally, within helpfulness, there are multiple dimensions like accuracy, relevance, and creativity, but they are all combined into a single metric in this study. Finally, even though our evaluation procedure can be extended to newer models, the rapid pace of advancements in the field could render the specific findings presented in this work outdated.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. 2022. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*.

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. 2024. A survey on evaluation of large language models. *ACM Trans. Intell. Syst. Technol.*, 15(3).

Yann Dubois, Balázs Galambosi, Percy Liang, and Tatsunori B Hashimoto. 2024. Length-controlled alpacaeval: A simple way to debias automatic evaluators. *arXiv preprint arXiv:2404.04475*.

Clémentine Fourrier, Nathan Habib, Alina Lozovskaya, Konrad Szafer, and Thomas Wolf. 2024. Open llm leaderboard v2. `https://huggingface.co/spaces/open-llm-leaderboard/open_llm_leaderboard`.

Harrison Lee, Samrat Phatale, Hassan Mansoor, Thomas Mesnard, Johan Ferret, Kellie Lu, Colton Bishop, Ethan Hall, Victor Carbune, Abhinav Rastogi, and Sushant Prakash. 2024. Rlaif vs. rlhf: Scaling reinforcement learning from human feedback with ai feedback. *Preprint*, arXiv:2309.00267.

Dawei Li, Bohan Jiang, Liangjie Huang, Alimohammad Beigi, Chengshuai Zhao, Zhen Tan, Amrita Bhattacharjee, Yuxuan Jiang, Canyu Chen, Tianhao Wu, Kai Shu, Lu Cheng, and Huan Liu. 2025. From generation to judgment: Opportunities and challenges of llm-as-a-judge. *Preprint*, arXiv:2411.16594.

Dawei Li, Shu Yang, Zhen Tan, Jae Young Baik, Sukwon Yun, Joseph Lee, Aaron Chacko, Bojian Hou, Duy Duong-Tran, Ying Ding, et al. 2024. Dalk: Dynamic co-augmentation of llms and kg to answer alzheimer's disease questions with scientific literature. *arXiv preprint arXiv:2405.04819*.

Xiaonan Li and Xipeng Qiu. 2023. Mot: Memory-of-thought enables chatgpt to self-improve. *arXiv preprint arXiv:2305.05181*.

Tian Liang, Zhiwei He, Wenxiang Jiao, Xing Wang, Yan Wang, Rui Wang, Yujiu Yang, Shuming Shi, and Zhaopeng Tu. 2023. Encouraging divergent thinking in large language models through multi-agent debate. *arXiv preprint arXiv:2305.19118*.

Yiqi Liu, Nafise Sadat Moosavi, and Chenghua Lin. 2023. Llms as narcissistic evaluators: When ego inflates evaluation scores. *arXiv preprint arXiv:2311.09766*.

Bhuvanashree Murugadoss, Christian Poelitz, Ian Drosos, Vu Le, Nick McKenna, Carina Suzana Negreanu, Chris Parnin, and Advait Sarkar. 2024. Evaluating the evaluator: Measuring llms' adherence to task evaluation instructions. *Preprint*, arXiv:2408.08781.

Yanxin Shen, Lun Wang, Chuanqi Shi, Shaoshuai Du, Yiyi Tao, Yixian Shen, and Hang Zhang. 2024. Comparative analysis of listwise reranking with large language models in limited-resource language contexts. *arXiv preprint arXiv:2412.20061*.

Lin Shi, Chiyu Ma, Weicheng Ma, and Soroush Vosoughi. 2024. Judging the judges: A systematic investigation of position bias in pairwise comparative assessments by llms. *Preprint*, arXiv:2406.07791.

Sijun Tan, Siyuan Zhuang, Kyle Montgomery, William Y Tang, Alejandro Cuadron, Chenguang Wang, Raluca Ada Popa, and Ion Stoica. 2024. Judgebench: A benchmark for evaluating llm-based judges. *arXiv preprint arXiv:2410.12784*.

Aman Singh Thakur, Kartik Choudhary, Venkat Srinik Ramayapally, Sankaran Vaidyanathan, and Dieuwke Hupkes. 2024. Judging the judges: Evaluating alignment and vulnerabilities in llms-as-judges. *Preprint*, arXiv:2406.12624.

Zhiyuan Zeng, Jiatong Yu, Tianyu Gao, Yu Meng, Tanya Goyal, and Danqi Chen. 2024. Evaluating large language models at evaluating instruction following. In *International Conference on Learning Representations (ICLR)*.

Lirui Zhao, Yue Yang, Kaipeng Zhang, Wenqi Shao, Yuxin Zhang, Yu Qiao, Ping Luo, and Rongrong Ji. 2024. Diffagent: Fast and accurate text-to-image api selection with large language model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6390–6399.

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. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Preprint*, arXiv:2306.05685.

A Additional Results

The **LLMBar**(Zeng et al., 2024) benchmark compares model responses to human-annotated datasets across two primary sets. The **Natural dataset** consists of real-world scenarios that evaluate models on typical instructions and responses. These examples are curated to ensure one response is objectively better, offering a reliable measure of the model's ability to follow instructions. This dataset reflects the model's performance on unmodified, naturally occurring prompts.

Conversely, the **Adversarial dataset** is designed to challenge LLM evaluators. It presents prompts where one response deviates from the instructions but may appear more appealing due to superficial qualities, such as tone or formatting. This dataset tests a model's robustness in adhering to instructions under more challenging conditions.

To enhance LLM evaluations within LLMBar, various prompting strategies are employed:

- **Vanilla**: The model is asked to select the better output based on the instruction, serving as a baseline strategy.
- **Chain-of-Thought (CoT)**: The model generates reasoning before making a decision, encouraging a deeper evaluation process.
- **Self-Generated Reference (Reference)**: The model first generates its own response, which is then used as a reference to evaluate other outputs.
- **Rules**: Explicit rules are provided to guide the model in prioritizing faithfulness to the instructions, improving performance across diverse contexts.
- **Metrics**: The model defines its own metrics for evaluating the quality of outputs, helping it focus on relevant aspects during assessment.
- **Swap and Synthesize (Swap)**: The model evaluates the outputs in both possible orders and synthesizes the results to mitigate positional bias.

Each strategy addresses specific challenges in LLM evaluation, with certain combinations often leading to improved performance.

The results in Table 2 demonstrate a clear performance gap between large and small models. Larger models consistently achieve higher accuracy, particularly in the more challenging adversarial dataset. Among smaller models, Gemma2:9B stands out with surprisingly strong performance, achieving scores comparable to larger models across both natural and adversarial tasks. The Metrics+Reference strategy generally yields the best results, suggesting that having models generate their own reference solutions improves evaluation quality.

The **Position Bias Analyzer** (Shi et al., 2024) assesses models' susceptibility to positional bias using datasets from MT-Bench (Zheng et al., 2023). Like LLMBar, this benchmark reverses the prompt order to evaluate how consistently models respond to varying prompt positions. Key metrics include *Positional Agreement*, which measures consistency before and after prompt reversal, and *Extraction Success Rate*, which evaluates the model's effectiveness in extracting relevant information. The benchmark also introduces the *Positional Preference Score*, ranging from -1 to 1, indicating the model's bias: a score of 0 denotes no bias, 1 indicates recency bias (favoring the first response), and -1 indicates primacy bias (favoring the last response).

The results show that larger models generally achieve better positional consistency and extraction success rates. Notably, Llama3.3:70B demonstrates the highest positional consistency (88.8%) among large models, while Qwen2.5:72B shows the most stable consistency with the lowest standard deviation (8.6). Most models exhibit relatively small positional biases, with scores close to 0, though some smaller models like Granite-3.1:8B show stronger primacy bias (-0.456). Gemma2:9B again performs remarkably well for its size, achieving perfect extraction rates and leading positional consistency among smaller models.

LLMBar - Natural and Adversarial																
	Natural								Adversarial							
	Vanilla		Metrics+Reference		Swap		Swap+CoT		Vanilla		Metrics+Reference		Swap		Swap+CoT	
Model	Acc	PA	Acc	PA	Acc	PA	Acc	PA	Acc	PA	Acc	PA	Acc	PA	Acc	PA
GPT4*	95.5%	95.0%	96.0%	96.0%	94.5%	97.0%	94.0%	100.0%	80.4%	91.5%	83.3%	89.5%	79.6%	96.2%	79.9%	97.3%
Llama3:70B	87.5%	85.0%	90.0%	88.0%	90.0%	94.0%	90.5%	93.0%	70.9%	82.4%	76.8%	84.0%	76.6%	90.1%	76.4%	91.6%
Llama3.1:70B	90.5%	91.0%	**95.5%**	93.0%	**95.0%**	**98.0%**	92.0%	96.0%	79.9%	82.1%	82.6%	84.9%	**84.7%**	92.1%	**83.2%**	93.1%
Llama3.3:70B			**95.5%**	95.0%							**83.3%**	87.1%				
Gemma2:27B	88.5%	**93.0%**	88.0%	90.0%	87.5%	93.0%	90.0%	96.0%	70.9%	81.5%	75.2%	81.1%	72.6%	95.1%	73.1%	94.6%
Qwen2:72B	**91.5%**	91.0%	93.5%	91.0%	92.5%	95.0%	**93.0%**	96.0%	70.1%	88.6%	73.5%	89.1%	72.6%	95.0%	72.2%	93.1%
Qwen2.5:72B	**91.5%**	**93.0%**	94.5%	93.0%	91.0%	92.0%	91.5%	93.0%	67.7%	82.3%	73.5%	86.4%	73.9%	89.6%	74.6%	87.5%
Mixtral:8x7B	82.5%	81.0%	82.5%	79.0%	85.0%	86.0%	82.0%	90.0%	57.3%	66.5%	58.7%	66.5%	55.3%	79.0%	58.9%	87.1%
Llama3:8B	75.5%	77.0%	81.0%	78.0%	77.5%	86.0%	78.5%	**96.0%**	41.8%	61.5%	56.1%	64.5%	45.0%	70.8%	46.4%	82.5%
Llama3.1:8B	81.0%	83.0%	80.0%	**85.5%**	83.0%	84.0%	**94.0%**	94.0%	45.1%	64.2%	55.2%	66.8%	55.9%	73.5%	56.2%	79.1%
Gemma2:9B	**87.5%**	**91.0%**	**90.0%**	**90.0%**	**90.5%**	**93.0%**	92.0%	92.0%	**71.1%**	**83.7%**	**74.5%**	**80.5%**	**66.2%**	**84.2%**	**66.7%**	**84.2%**
Qwen2:7B	83.0%	82.0%	83.5%	79.0%	80.5%	87.0%	80.0%	88.0%	47.2%	61.6%	54.8%	55.7%	41.5%	78.9%	40.0%	79.0%
Qwen2.5:7B	**88.5%**	85.0%	89.5%	87.0%	85.5%	**93.0%**	88.0%	94.0%	63.4%	66.5%	65.9%	74.5%	60.3%	82.9%	59.7%	**88.3%**
Mistral:7B	67.5%	37.0%	68.5%	41.0%	78.0%	67.0%	79.5%	77.0%	44.8%	28.3%	47.4%	28.6%	43.5%	59.0%	49.0%	63.2%
Granite-3.1:8B	82.0%	78.0%	79.5%	77.0%	**89.0%**	90.0%	85.5%	92.0%	51.3%	61.1%	56.5%	59.1%	51.3%	61.1%	51.3%	61.1%
Phi-3-small	86.0%	84.0%	89.5%	87.0%	87.0%	94.0%	88.5%	93.0%	53.8%	72.4%	61.3%	74.5%	53.8%	72.4%	53.8%	72.4%
Falcon3:7B	81.5%	75.0%	83.5%	81.0%	84.5%	**93.0%**	78.0%	88.0%	45.1%	76.6%	51.8%	73.9%	45.1%	76.6%	45.1%	76.6%
Internlm2.5:7b	81.0%	71.0%	82.0%	75.0%	80.0%	76.0%	83.0%	85.0%	50.6%	60.5%	57.3%	65.6%	50.6%	60.5%	50.6%	60.5%

Table 2: Performance comparison of language models on the LLMBar benchmark across Natural and Adversarial datasets. Results show accuracy (Acc) and positional agreement (PA) scores for different evaluation strategies: Vanilla, Metrics+Reference, Swap, and Swap+CoT. Models are grouped by size, with larger models (>20B parameters) shown above the line and smaller models below. Bold values indicate top performance within each group. *We source GPT-4-0613 results from (Zeng et al., 2024).

Position Bias Analyzer - MT-Bench				
	Extraction Successful Rate	Positional Consistency	Positional Consistency Std	Positional Preference Score
GPT4*	100.0%	81.5%	14.9	0.020
Llama3:70B	**100.0%**	83.8%	11.1	-0.025
Llama3.1:70B	**100.0%**	83.1%	15.3	0.006
Llama3.3:70B	**100.0%**	**88.8%**	12.2	-0.075
Gemma2:27B	**100.0%**	71.9%	18.1	0.044
Qwen2:72B	**100.0%**	79.4%	15.2	-0.131
Qwen2.5:72B	**100.0%**	84.4%	**8.6**	-0.056
Mixtral:8x7B	95.8%	76.2%	11.8	0.019
Llama3:8B	84.8%	62.5%	18.8	-0.131
Llama3.1:8B	98.7%	73.1%	13.5	-0.125
Gemma2:9B	**100.0%**	**71.2%**	**11.1**	**0.175**
Qwen2:7B	98.7%	46.8%	17.8	0.069
Qwen2.5:7B	98.7%	38.5%	15.9	-0.188
Mistral:7B	**100.0%**	58.1%	19.4	-0.256
Granite-3.1:8B	97.8%	40.5%	18.2	-0.456
Phi-3-small	97.8%	56.9%	21.8	-0.188
Falcon3:7B	99.7%	53.5%	17.0	-0.150
Internlm2.5:7b	51.0%	57.1%	30.9	0.056

Table 3: Results from the Position Bias Analyzer benchmark on MT-Bench data. Models are evaluated on their ability to extract information (Extraction Success Rate), maintain consistent responses across prompt reversals (Positional Consistency), the standard deviation of consistency (Positional Consistency Std), and their positional bias tendency (Positional Preference Score). Models are grouped by size, with larger models (>20B parameters) above the line and smaller models below. Bold values indicate top performance within each group. *We respectively source GPT-4-0613 results from (Shi et al., 2024).