Mitigating Self-Preference by Authorship Obfuscation

Taslim Mahbub, Shi Feng

George Washington University
taslim.mahbub,shi.feng@gwu.edu

Abstract

Language models (LMs) judges are widely used to evaluate the quality of LM outputs. Despite many advantages, LM judges display concerning biases that can impair their integrity in evaluations. One such bias is self-preference: LM judges preferring their own answers over those produced by other LMs or humans. The bias is hard to eliminate as frontier LM judges can distinguish their own outputs from those of others, even when the evaluation candidates are not labeled with their sources. In this paper, we investigate strategies to mitigate self-preference by reducing the LM judges' ability to recognize their own outputs. We apply black-box perturbations to evaluation candidates in pairwise comparison to obfuscate the authorship and reduce self-recognition. We find that perturbations as simple as synonym replacement for a few words predictably reduce self-preference. However, we also uncover fundamental challenges to eliminating the bias: when we extrapolate our perturbations to a more complete neutralization of stylistic differences between the evaluation candidates, self-preference recovers. Our findings suggest that self-recognition and self-preference can happen on many semantic levels, and complete mitigation remains challenging despite promising initial results.

Code —
https://github.com/Taslim-M/mitigatingselfpreference
Extended version — https://arxiv.org/pdf/2512.05379

Introduction

Language models (LMs) are frequently used in place of human evaluators to judge the quality of LM outputs. These LM judges are now widely used for benchmarking (Zheng et al. 2023; Bai et al. 2023), to help train LMs as reward models (Leike et al. 2018; Stiennon et al. 2020), and to guide inference-time compute (Saunders et al. 2022; Chen et al. 2023). This "LM-as-a-judge" approach has clear scalability advantages, but also suffers from biases that can hurt the integrity of evaluations and derail model development. One particularly concerning bias is self-preference (Zheng et al. 2023; Liu, Moosavi, and Lin 2023; Panickssery, Bowman, and Feng 2024; Li et al. 2025; Goel et al. 2025; Ohi et al. 2024; Chen et al. 2025), where the LM judge prefers its own answers over those produced by other models or humans, even when the alternative's quality is objectively higher. Self-preference is particularly harmful when the judge prefers its own incorrect or unsafe answers over correct answers from other models (Chen et al. 2025). Judges with harmful self-preference can amplify untruthfulness and deceitfulness, a concern for both capability and safety.

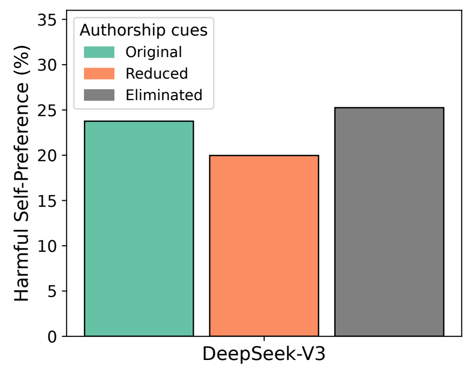

Figure 1: Harmful self-preference of DeepSeek-V3 in pairwise comparison. When we perturb the answer pair to reduce superficial cues (e.g., word choice) that the judge can use to infer model identity, harmful self-preference also reduces. However, the effect is reversed when we eliminate identity cues by paraphrasing.

In this paper, we investigate strategies to mitigate harmful self-preference. We build on the self-recognition hypothesis of Panickssery, Bowman, and Feng (2024): self-preference is partially caused by the judges being able to recognize themselves. Following this hypothesis, we seek to mitigate self-preference by reducing the judge's ability to distinguish its own outputs from others: if the judge cannot tell which answer is its own, it's less likely to be biased. We do so by applying black-box perturbations to evaluation candidates to obfuscate their authorship.

We show that perturbations as simple as synonym replacement for a few words can indeed reduce harmful self-preference. This suggests that self-recognition and self-preference is indeed reliant on superficial, stylistic cues such as word choice. However, we also discover that a complete mitigation by black-box perturbations is challenging: when we apply further perturbations to neutralize the stylistic cues, self-preference recovers. The reason why this happens, as we design experiments to validate, is that self-recognition and self-preference happen on many semantic levels: when stylistic cues are removed, the judge can still distinguish their own answer from others when they express different opinions. This observation reveals the impossibility of a complete elimination of self-preference, which would require the judge's decisions to be decoupled from their beliefs about what is correct. A more plausible framing of the research question, as we discuss, should focus on preventing the judge from overly relying on its prior belief.

Evaluating Harmful Self-preference

We focus on pairwise comparison, a common format of using LM as a judge for benchmarking (Chiang et al. 2024) and reward modeling (Stiennon et al. 2020). Given answers from two LMs, the LM judge picks the better one according to criteria given in the prompt. When one of the LMs being evaluated is the same as the judge, we say that the judge is performing a self-evaluation.[1]

We define self-preference as the judge selecting their own answer in self-evaluation. Such a preference is harmless if the judge's answer is indeed the better one, but harmful if otherwise. On tasks where answer quality can be objectively determined (e.g., by expert annotation), we can label self-preference as harmful when the judge selects their own answer when the competitor's answer is objectively better.

Modeling

We evaluate self-preference using five instruction-tuned LMs of varying sizes and capabilities: (1) Llama-3.1-8B, (2) Qwen2.5-7B, (3) Llama-4-Scout-17B, (4) Llama-4-Maverick-17B, and (5) DeepSeek-V3-37B. We set the temperature to zero for all judge decisions.

Testbed: Long-document QA

To evaluate objective answer quality, we rely on multiple-choice questions with human-annotated correct answers. In particular, we use the QuALITY dataset (Pang et al. 2021), which consists of long-form English passages (averaging around 4200 words) accompanied by multiple-choice questions. The passages are drawn from fictional narratives and magazine articles, designed to evaluate LM's comprehension of long-form texts. Each passage is followed by a question with four answer choices. We use the validation set of QuALITY, which contains 2,086 examples.

For each of the two models we are comparing, we provide the passage, the question, followed by the answer choices. We instruct the model to select an option from the list and provide a reasoning for the answer. We refer to these two parts of the model output as *label choice* and *reasoning*. We instruct the judge to evaluate the two models based on both their reasoning and their label choices, and provide context, including the passage and the question. Finally, evaluate the judge's decision based on whether the selected label choice is correct. Doing so allows us to objectively evaluate the judge and to label self-preference as harmful or harmless.

To control for ordering bias, we query the judge twice for each pairwise comparison, where the two evaluation candidates are swapped. A decision by the judge is only considered valid if it remains consistent across these two queries; otherwise, the judge is considered ambiguous. We report the frequency that a judge is ambiguous, and remove those decisions from subsequent analysis.

After removing ambiguous decisions, the judge's correctness—and subsequently the harmlessness of its self-preference—can still be ambiguous. Some answer pairs might contain two correct answers or two incorrect answers—or, more generally, contain two answers of objectively comparable quality. We also omit these examples from subsequent analysis.

To summarize, we use the following process to construct a dataset of answer pairs for our analysis: we first gather all models' answers on validation examples of QuALITY. Then, for every pair of models, we remove answer pairs with comparable answers. We then use every model as a judge to evaluate all remaining answer pairs, report the frequency of ambiguous decisions, and remove them. In this collection of answer pairs and corresponding judge decisions, we can always use the groundtruth label to objectively determine if the judge's decision is correct or not. We refer to this dataset as "all examples".

Stronger Models Are More Accurate Judges

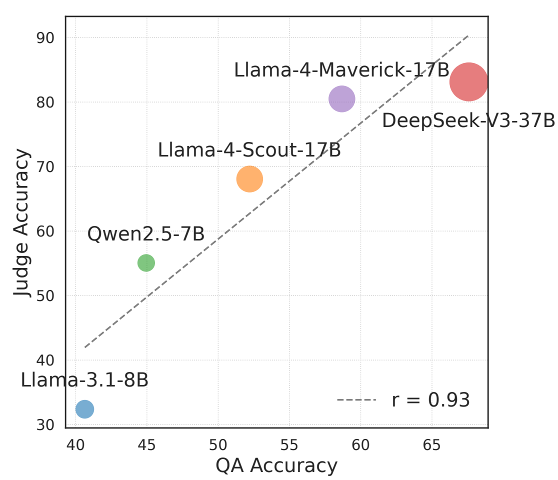

Figure 2: Bigger models are more accurate at both answering questions and judging.

Bigger models are generally more accurate at both answering questions and judging (Figure 2) and make fewer ambiguous decisions (Figure 3).

In our experiments, we also find that more decisive models tend to be more accurate as judges (Figures 2 and 3),

[1]For simplicity, we say that the judge is comparing its *own* answer against a competitor. But we should note that the model receives different prompts in its two roles and does not behave exactly the same.

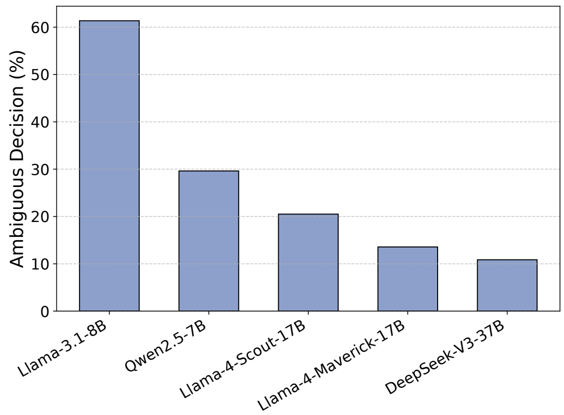

Figure 3: Capable models are less sensitive to the order of evaluation and make fewer ambiguous decisions.

suggesting that when a model is confident about an answer selection, the decision is more likely to be correct.

In the following analysis, we omit all ambiguous answer pairs and judge decisions, such that for every pairwise comparison, we can definitively label the judge's decision as correct or incorrect. Because judges might not be ambiguous on the same set of examples, this omission leads to some variations in the answer pairs that each judge is evaluated on.

Stronger Models Overestimate Their Accuracy

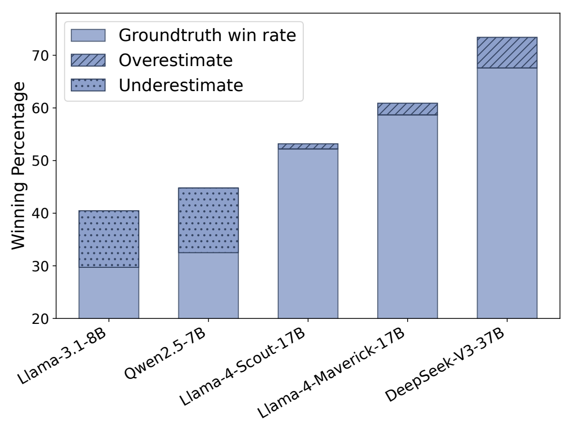

Figure 4: Win rate of each model against all others as judged by the groundtruth compared to the model itself. Stronger models overestimate their accuracy; weaker models do the opposite.

For every model, we calculate its win rate against all other models: the percentage of examples where the model provides an objectively better answer than another model. We can compare the win rate calculated based on the groundtruth with the model's own estimate to characterize self-preference. As we see in Figure 4, stronger models significantly overestimate their own accuracy. In particular, according to DeepSeek-V3's own estimation, its win rate against other models is 73%, but in reality, its win rate is 66%. Conversely, weaker models underestimate their accuracy: they frequently prefer other models' reasoning, even when their own answer is correct.

Harmful Self-Preference in Large Models

To further isolate the impact of harmful self-preference, we zoom in on each judge's accuracy in self-evaluation and in the harmful cases—the subset of self-evaluation where the judge's answer is wrong. As shown in Figure 5, the accuracy of more capable judges takes a bigger dive in the harmful cases, suggesting more significant harmful self-preference. In particular, DeepSeek-V3 is more accurate than LLama-4-Scout-17B overall and in self-evaluation, but its degradation on harmful cases is much more significant, making it less accurate than the two LLama-4 models in those cases. This indicates that DeepSeek-V3, as a judge, shows a stronger tendency to stick to its own reasoning when it's wrong—it is more "egotistical". Harmful self-preference is distinctively a tendency that occurs in the three larger models: although they are overall more accurate judges, we can trust them less to spot their own mistakes, even when a better alternative is presented.

Mitigating Harmful Self-preference

The previous section validates that despite higher accuracy, the more capable judge also exhibits harmful self-preference. This is concerning: in general, we will always want to use the most capable and accurate judge, but the harmful self-preference also means that we need to be more cautious with them. In this section, we investigate strategies to mitigate the self-preference bias, and empirically examine whether there is a trade-off between accuracy and bias. We base our study on the self-recognition hypothesis that self-preference is partly driven by the judge's ability to differentiate their own answers from others. Motivated by how LM detection methods (Mitchell et al. 2023) exploit the model's sensitivity to small changes in the input, we hypothesize that self-recognition is also sensitive to small perturbations, which means we can mitigate self-preference by reducing self-recognition using small perturbations.

Validating the Self-Recognition Hypothesis

We begin our investigation by validating the connection between self-preference and self-recognition. Following Panickssery, Bowman, and Feng (2024), we prompt the judge to identify which of the two evaluation candidates it believes to have been generated by itself, in a context separate from self-evaluation. In the subset of harmful cases, there was a positive correlation between self-recognition and self-preference, $r(6667) = 0.63$, $p < .001$, 95% CI [0.62, 0.65]. A chi-square test further finds a statistically significant relationship between them (p <0.001, df = 1). This indicates that models are substantially more likely to favor incorrect responses that they recognize as their own.

Perturbation by Synonym Replacement

Based on our validation that self-preference is associated with self-recognition, we explore strategies to mitigate self-

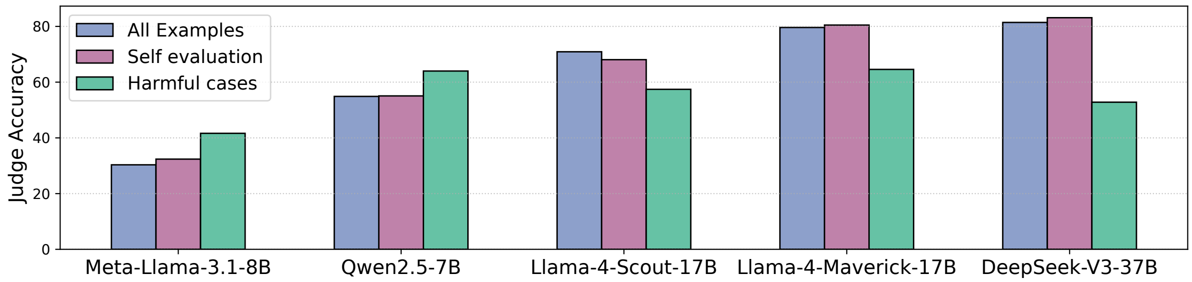

Figure 5: Strong models are significantly less accurate on examples where their own answers are wrong (harmful cases), but have a higher overall judge accuracy.

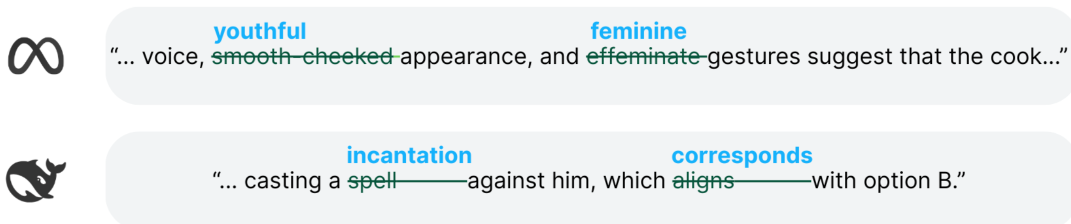

Figure 6: Examples of synonym replacement (top, Llama-3.1 and bottom DeepSeek-V3). Objective answer quality is not affected.

preference by obfuscating authorship: if the judge cannot recognize the answer as its own, it wouldn't favor it disproportionately.

One key intuition drawn from the LM detection literature is that LMs are more sensitive to perturbations when they are applied to texts that are generated by the same LM than others. Zero-shot methods (Mitchell et al. 2023; Bao et al. 2023) achieve high detection accuracy by drawing a threshold on the perplexity change when small perturbations are applied to the text being evaluated: if the perturbation leads to a large change in perplexity, it's likely that the text was generated by the LM. Following this intuition, we hypothesize that self-recognition is also highly dependent on stylistic cues such as word choices, and that small perturbations with no affect on the semantic meaning are sufficient for reducing self-recognition.

The first perturbation we experiment with is synonym replacement: in self-evaluation, we replace two words in *the judge's answer* with their synonyms. We anticipate that this would reduce the judge's ability to discern which of the two evaluation candidate is its own generation, and subsequently reduce self-preference.

We prompt a LLaMA-3.3-70B (not in our judge pool) to identify replacement candidates in the reasoning that are neither stop words nor present in the question, to avoid disruptive changes and ensure the relevance of the answer remain unaffected. We use a third-party model for this purpose to avoid bias against any judge. This process produces subtle word changes that do not affect answer quality or relevance to the question (Figure 6). We manually verify that by looking at the perturbed answer, we cannot tell which word has

Figure 7: Two-word synonym replacement reduces overall self-recognition, especially for the more capable models.

been replaced; even when comparing the original and perturbed versions side by side, it is not possible to determine which version is which. The number of words to replace is a hyperparameter that we leave for future work to explore.

Synonym Replacement Reduces Self-Recognition and Self-Preference

We repeat the self-recognition evaluations on the harmful cases and observe a significant reduction in self-recognition after perturbing only two words (Figure 7), with a more pronounced impact on the stronger models. This confirms our hypothesis that self-recognition is sensitive to superficial,

Figure 8: Synonym replacement reduces harmful self-preference and increases the accuracy of judges when their own answer is wrong. A paired t-test revealed a significant difference in preference between the original and perturbed conditions ($p < .01$).

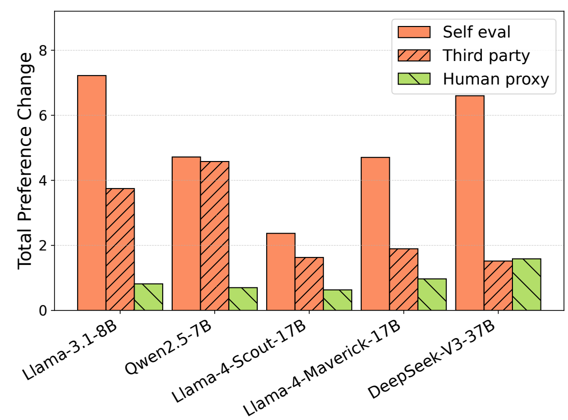

Figure 9: The impact of perturbation measured by total preference change is significantly higher on judges in self-evaluation than as a third-party. Human proxy also confirms that perturbation has a minimal effect on answer quality.

stylistic cues like word choices, and that black-box perturbations can effectively obfuscate authorship.

We then validate that perturbations achieve the expected effect of mitigating self-preference. Figure 8 shows a consistent increase in the judge's accuracy on the harmful cases—examples where the judge's answer is objectively the worse one. This means that after perturbation, the judges are indeed less likely to prefer their own incorrect answers—a reduction in harmful self-preference.

Judges Are More Affected by Perturbations in Self-Evaluation Than as a Third-Party

Despite our effort to minimize the impact of synonym replacements on the answer's semantic qualities, we cannot rule out that it contributes to the reduction of self-preference, that the perturbed answer is, in fact, lower quality. In other words, the perturbations can lower self-preference through two paths: by reducing self-recognition as we desire, and by damaging the answer's quality. To control for the latter, we examine the impact of perturbation on each judge as a self-evaluating judge and as a third-party judge. To be more concrete, we first compute the total change in preference of all judges in self-evaluation, as we do for Figure 8; we then use each judge to evaluate answer pairs generated by two other models, and compute the total preference before and after perturbation. The underlying answer pairs (including the perturbed versions) remain the same under these two conditions; the only variable is whether the judge's own answer is being perturbed. As we see in Figure 9, all judges are significantly more sensitive to perturbations in self-evaluation than as a third-party—although synonym replacement might have a small impact on objective answer quality, the reduction in self-preference is mainly caused by reducing self-recognition. We additionally validate that perturbations have minimal effect on objective quality using a frontier commercial model (o3) as an approximation for human judgment. To complement this, we also conduct a human study on 500 examples, confirming that human raters judged the original and perturbed versions as equivalent in quality.

Perturbation by Judge-Paraphrasing

Our experiments with synonym replacement validate that stylistic cues such as word choices are indeed the basis of self-recognition, and that by perturbing those, we can reduce self-preference. So we take it further: we use the judge to paraphrase the competitor's answer so they both look like they were produced by the judge. If our logic holds, this should completely neutralize these superficial cues and further reduce self-preference. We call this perturbation *judge paraphrasing*, and Figure 10 presents a few examples.

Judge Paraphrasing Increases Self-Recognition and Self-Preference

Judge paraphrasing achieves the opposite effect than we expected: it increases self-recognition (Figure 11), and increases harmful self-preference (Figure 12). This means that, when the stylistic differences are removed, the judge is relying on semantic differences between the two answers to judge which one was written by itself and which one to favor. In retrospect, this makes sense: now that style is no longer a factor, it is simply choosing the answer that it "agrees" with.

Contrasting the two perturbations, we see that self-recognition and self-preference can occur at multiple semantic levels: it can prefer an answer because its style resembles the judge's own, or because it expresses an opinion that the judge agrees with. Based on our experiments, when *superficial self-recognition* is removed, *shared belief* takes over as the driving force behind self-preference.

To examine the dynamics between these two factors, we cross-examine each judge with answer pairs where these two factors act against each other: the label/option chosen by the judge paired with a reasoning generated by the competitor,

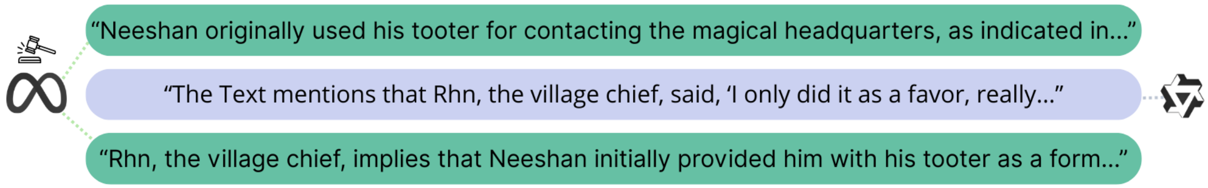

Figure 10: Examples of judge paraphrasing. We prompt the judge (in example, Llama-4-Scout) to paraphrase the competitor's (Qwen-2.5) answer while maintaining semantics, so that both evaluation candidates look like they were produced by the judge in terms of style.

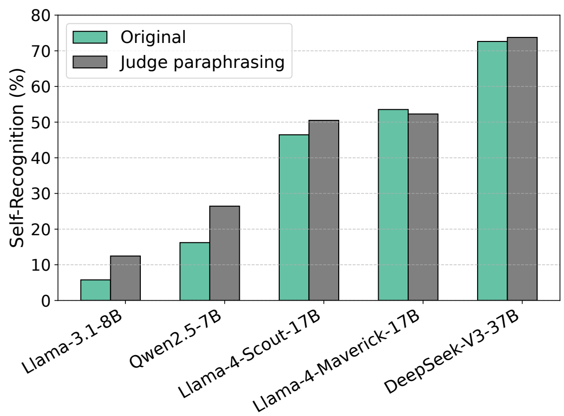

Figure 11: Using the judge to paraphrase the competitor—which should neutralize all stylistic cues—ends up increasing self-recognition.

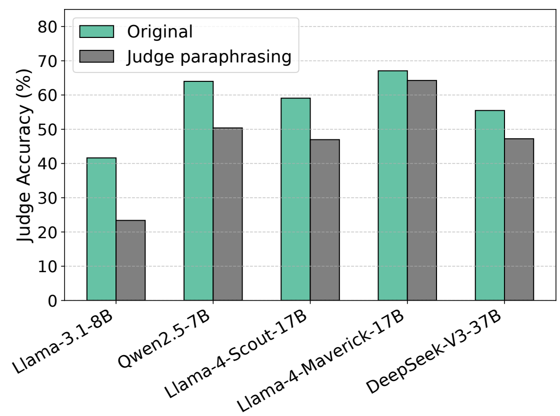

Figure 12: Judge paraphrasing decreases judge accuracy in the harmful cases and exacerbates harmful self-preference.

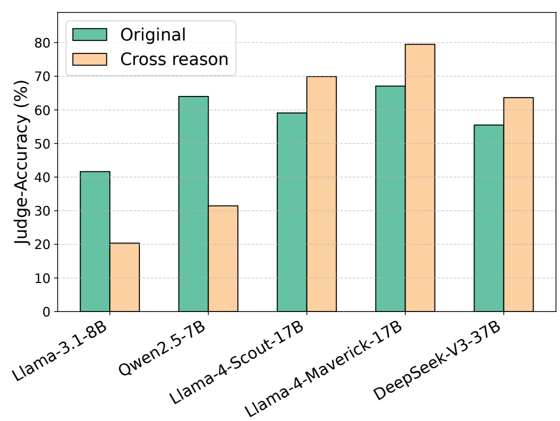

Figure 13: Using the judge to generate a reason for the competitor's answer option and vice versa yields 2 observations: weaker models tend to prefer the competitor's reasoning more, and stronger models tend to choose their own reasoning.

and vice versa. In other words, the judge would see one answer expressing an opinion that it agrees with, and another answer with an opposing opinion, but in a style it's more familiar with. If, after this intervention, the judge changes its decision to favor the answer with a familiar-looking style, we can conclude that superficial self-recognition has a higher weight in that decision.

Figure 13 illustrates the two contrasting outcomes observed in the cross-reason experiments. For the weaker models, we observe a negative effect of shared belief, where incorporating the competitor's reason reinforces the model's original, incorrect answer, ultimately decreasing judge accuracy. In contrast, all three stronger models exhibit a positive self-recognition effect: they are more inclined to select the competitor's (correct) answer when it is paired with an explanation generated by themselves. This observation reinforces our finding that stronger models have a higher self-recognition ability and prefer answers written by them, even if it goes against their original selection.

Mitigating Self-Preference in Coding Task

To examine whether our mitigation of self-recognition and self-preference extends beyond a single task, we use the MBPP+ dataset from (Chen et al. 2025) to test code-level obfuscation techniques. Specifically, we use coding solutions generated by Llama-3.1-70B and Llama-3.3-70B, then evaluated against outputs from six other models. We rewrite the original code to introduce idiomatic style variations (e.g., converting imperative structures into declarative ones) while ensuring functional equivalence by matching the original code for all test cases provided in the question. Figure 14 illustrates the overall trend observed across both judge models: introducing minor stylistic variations while preserving the underlying code logic consistently reduces self-recognition ability and improves judge accuracy.

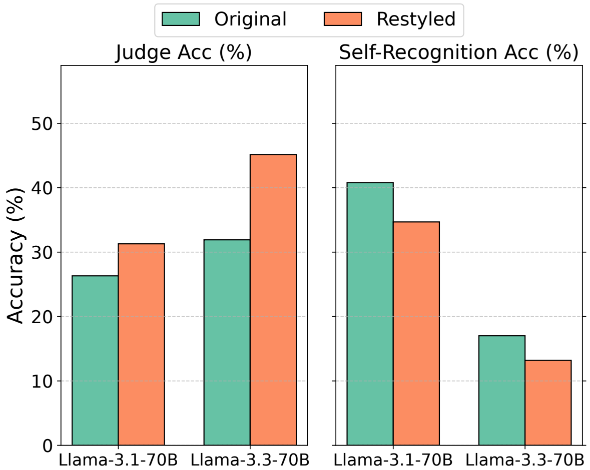

Figure 14: Restyling the code improves the judge-accuracy in harmful cases, while reducing self-recognition ability.

Conclusion, Discussion, and Limitations

We study self-preference in LM judges using QuALITY, a long-document question answering dataset. We find that the more capable models exhibit significant harmful self-preference: although they are generally more accurate, we need extra caution when tasking them to spot their own mistakes, even when objectively better options are presented. We validate the contribution of self-recognition to such bias, and present viable mitigation strategies using inference-time perturbations. At the same time, we illustrate the complexity of this task: self-recognition and self-preference can occur at multiple semantic levels, and a complete elimination is not plausible—developing mitigation strategies requires a more nuanced approach to formulate the objective.

White-box mitigation Recent work explores white-box methods such as activation steering to manipulate self-recognition (Ackerman and Panickssery 2024) and self-preference (Nguyen et al. 2025). We focus on black-box methods for two main reasons. First, the processes are less dependent on having access to specific models' internals and are more applicable as a standardization process for benchmarks. Second, the perturbations allow us to confirm the hypothesis that self-recognition and subsequently self-preference are indeed sensitive to superficial cues. An important future work would be to compare white-box and black-box methods for their robustness and flexibility, both of which are crucial to the proper debiasing of LM judges.

Models choose what they think is theirs Our findings suggest a strong connection between self-recognition and harmful self-preference—models tend to favor the output they believe they generated. Weaker models generally struggle with self-recognition and, as a result, exhibit weaker self-preference. This limitation can be advantageous: for instance, our experiments show that Qwen2.5 performs on par with models nearly five times larger in lose-case evaluations precisely because it does not demonstrate harmful self-preference.

Recommendations for LLM-as-a-Judge Systems We suggest several strategies to enhance the reliability of LLM-based judge systems. First, since models tend to favor their own outputs, excluding the evaluated model from serving as a judge can help reduce this bias. Additionally, targeted token perturbations can eliminate superficial self-identifying cues without compromising the overall quality of the response. We encourage future works to explore other methods of authorship obfuscation. Among the most effective approaches, employing a majority-vote ensemble of decisive judges offers a robust means of mitigating preference bias and improving judgment fairness.

Harmful Self-Preference and scalable oversight LLM-based evaluations is a key to achieving scalable oversight in many complex domains, where human supervision is costly and limited (specialized). A stronger model may be used as a supervisor in many tasks. Our experiments show that stronger models are able to recognize themselves more frequently, and suffer the most in loss cases due to harmful self-preference. Thus, supervisor models in scalable oversight problems must be explored in these tasks, where the models are not as effective as they are likely to provide biased or incorrect supervision. One such example is using an LLM as a monitor for reward hacking or scheming output, where using the same model for the detection of "cheating" may result in undetected examples.

Limitations

Controlling for Confounding Biases. While we focus on an objective task setup—where each evaluator is presented with a single correct answer and is instructed to judge based on the accompanying rationale—we do not explicitly control for factors such as persuasiveness, stylistic tone, or length bias that may influence the model's decision. These attributes can unintentionally sway preferences, even when the reasoning is logically sound. As a result, our analysis captures a potential interplay between subjective and objective factors. Future work should explore controlled manipulations of these stylistic elements to more rigorously assess the fairness and robustness of LMs in judgment roles.

Broader Impacts of Perturbation. Our study primarily targets the mitigation of self-preference bias in loss-case scenarios. While the perturbation techniques we evaluate are effective in this context, their impact on win-cases and on settings where both reasons are perturbed remains unexplored. A more comprehensive evaluation across diverse scenarios is necessary to determine whether these methods can serve as a general strategy for addressing superficial self-recognition and associated biases in model judgment.

References

Ackerman, C.; and Panickssery, N. 2024. Inspection and Control of Self-Generated-Text Recognition Ability in Llama3-8b-Instruct. *arXiv preprint arXiv:2410.02064*.

Bai, Y.; Ying, J.; Cao, Y.; Lv, X.; He, Y.; Wang, X.; Yu, J.; Zeng, K.; Xiao, Y.; Lyu, H.; et al. 2023. Benchmarking foundation models with language-model-as-an-examiner. *Advances in Neural Information Processing Systems*, 36: 78142–78167.

Bao, G.; Zhao, Y.; Teng, Z.; Yang, L.; and Zhang, Y. 2023. Fast-detectgpt: Efficient zero-shot detection of machine-generated text via conditional probability curvature. *arXiv preprint arXiv:2310.05130*.

Chen, W.-L.; Wei, Z.; Zhu, X.; Feng, S.; and Meng, Y. 2025. Do LLM Evaluators Prefer Themselves for a Reason? *arXiv preprint arXiv:2504.03846*.

Chen, X.; Lin, M.; Schärli, N.; and Zhou, D. 2023. Teaching large language models to self-debug. *arXiv preprint arXiv:2304.05128*.

Chiang, W.-L.; Zheng, L.; Sheng, Y.; Angelopoulos, A. N.; Li, T.; Li, D.; Zhu, B.; Zhang, H.; Jordan, M.; Gonzalez, J. E.; et al. 2024. Chatbot arena: An open platform for evaluating llms by human preference. In *Forty-first International Conference on Machine Learning*.

Goel, S.; Struber, J.; Auzina, I. A.; Chandra, K. K.; Kumaraguru, P.; Kiela, D.; Prabhu, A.; Bethge, M.; and Geiping, J. 2025. Great Models Think Alike and this Undermines AI Oversight. arXiv:2502.04313.

Leike, J.; Krueger, D.; Everitt, T.; Martic, M.; Maini, V.; and Legg, S. 2018. Scalable agent alignment via reward modeling: a research direction. *arXiv preprint arXiv:1811.07871*.

Li, D.; Sun, R.; Huang, Y.; Zhong, M.; Jiang, B.; Han, J.; Zhang, X.; Wang, W.; and Liu, H. 2025. Preference Leakage: A Contamination Problem in LLM-as-a-judge. *arXiv preprint arXiv:2502.01534*.

Liu, Y.; Moosavi, N. S.; and Lin, C. 2023. LLMs as narcissistic evaluators: When ego inflates evaluation scores. *arXiv preprint arXiv:2311.09766*.

Mitchell, E.; Lee, Y.; Khazatsky, A.; Manning, C. D.; and Finn, C. 2023. DetectGPT: Zero-shot machine-generated text detection using probability curvature. In *International conference on machine learning*, 24950–24962. PMLR.

Nguyen, M.; Barzdukas, J.; Bozoukov, M.; and Fu, H. 2025. Manipulating Self-Preference in LLMs. https://www.lesswrong.com/posts/a5Eb8JxkGuASMaZWk/manipulating-self-preference-in-llms. LessWrong blog post, July 1, 2025.

Ohi, M.; Kaneko, M.; Koike, R.; Loem, M.; and Okazaki, N. 2024. Likelihood-based mitigation of evaluation bias in large language models. *arXiv preprint arXiv:2402.15987*.

Pang, R. Y.; Parrish, A.; Joshi, N.; Nangia, N.; Phang, J.; Chen, A.; Padmakumar, V.; Ma, J.; Thompson, J.; He, H.; and Bowman, S. 2021. QuALITY: Question answering with long input texts, yes! In *NAACL*.

Panickssery, A.; Bowman, S.; and Feng, S. 2024. LLM evaluators recognize and favor their own generations. *Advances in Neural Information Processing Systems*, 37: 68772–68802.

Saunders, W.; Yeh, C.; Wu, J.; Bills, S.; Ouyang, L.; Ward, J.; and Leike, J. 2022. Self-critiquing models for assisting human evaluators. *arXiv preprint arXiv:2206.05802*.

Stiennon, N.; Ouyang, L.; Wu, J.; Ziegler, D.; Lowe, R.; Voss, C.; Radford, A.; Amodei, D.; and Christiano, P. F. 2020. Learning to summarize with human feedback. *Advances in neural information processing systems*, 33: 3008–3021.

Zheng, L.; Chiang, W.-L.; Sheng, Y.; Zhuang, S.; Wu, Z.; Zhuang, Y.; Lin, Z.; Li, Z.; Li, D.; Xing, E.; et al. 2023. Judging LLM-as-a-judge with MT-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36: 46595–46623.