Proceedings of the 4th International Conference on Mathematical Physics and Computational Simulation
DOI: 10.54254/2753-8818/2026.33416

Analysis of LLM-as-a-Judge Position Bias Mechanism Based on Controlled Synthesis Data

Zhiyin Yang

School of Economics and Management, Communication University of China, Beijing, China
202319263003@mails.cuc.edu.cn

Abstract. With the widespread application of large language models in automatic evaluation tasks, Large Language Model (LLM)-as-a-Judge has gradually become an important paradigm for reward modeling and model alignment. Existing research mainly improves judgment ability through thought chain reasoning and reinforcement learning, but pays less attention to the impact of training data distribution structure characteristics on model learning behavior. Addressing the positional bias problem in large language models' generated answers, this paper constructs a controllable synthetic preference data environment from the perspective of data distribution and compares the learning differences of reward models under biased and unbiased training mechanisms. The experiment uses a logistic regression model as the reward function approximator and analyzes the model's accuracy, positional bias rate, and parameter structure changes while keeping the test set unbiased. The results show that when there is a statistical correlation between order and label in the training data, the positional feature weight is 1.047, significantly higher than the control group's weight of 0.303. This indicates that the model significantly increases its dependence on positional features, and even if the overall performance does not decrease significantly (94.67% & 98.33%), its internal decision-making mechanism still shifts. The study reveals that data structure features may alter the model's discrimination criteria without affecting surface performance, providing a mechanism-level reference for reward model design and data construction.

Keywords: LLM-as-a-Judge, Reward model, Position bias, Reinforcement learning, Structural shortcut learning

1. Introduction

In recent years, significant progress has been made in the research of large language models as judges. Existing work mainly proceeds in two directions: First, improving the quality of evaluation through thought chain reasoning [1]. Saha et al. proposed EvalPlanner, which decouples evaluation into planning and execution phases [2]; Lambert et al. constructed the RewardBench benchmark, providing a standardized platform for reward model evaluation [3]; Wang et al. proposed Self-Taught Evaluator, proving that model performance can be iteratively improved without manual annotation [4]; Zhang et al. introduced group comparison reasoning, stimulating more detailed evaluation through diverse references [5]; Yu et al. improved the performance and practicality of

© 2026 The Authors. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

DOI: 10.54254/2753-8818/2026.33416

Large Language Model as a Judge through improved SFT-DPO training and efficient data strategies [6]. Second, introducing reinforcement learning into the training of judging models [1]. The J1 work of the Meta FAIR team first generated long thought chains through the GRPO incentive model [7], and Wang & Li et al. further extended it to the multimodal domain, verifying the universality of long chain reasoning [8].

However, the above studies mainly focus on improving the reasoning ability of the evaluation model, and pay less attention to the impact of training data structure features on the model's learning behavior. Zhou et al.'s research on shortcut convex hull learning shows that statistical correlation in the data will lead to the model learning task-irrelevant features [9]; Li and Gao revealed that the anchoring bias of GPT-2 is a typical example of the correlation between order and label [10]; Yu et al.'s review also pointed out that the design of reward models should fully consider the influence of data distribution [11].

Therefore, this study starts from the data distribution structure, designs two training environments, biased and unbiased, and systematically analyzes the impact of order correlation on the learning mechanism of reward models. The following hypothesis is proposed: when there is statistical correlation between order and label in the training data, the model will tend to use order features for prediction, and even if the test data remains unbiased, the internal decision structure of the model may still change.

2. Dataset description

This study constructed a controlled simulated dataset to investigate the impact of positional bias on reward model training. Each sample contains two candidate answers (denoted as Answer A and Answer B) for a given question. The model task is to determine which answer is of higher quality.

A random quality score q_A, q_B are generated for each answer, ranging from 0 to 1 and following a standard normal distribution. The true label is determined by the difference in quality scores: when $q_A > q_B$, the label is $y = 1$; otherwise, $y = 0$. This setting ensures that the labels have a clear causal source and avoids data contamination.

$$y_{true} = 1\left(q_A > q_B\right) \tag{1}$$

$$y_{train} = \begin{cases} 1 & \text{if flip condition satisfied} \\ y_{true} & \text{otherwise} \end{cases} \tag{2}$$

To simulate the potential order bias in real-world Large Language Model (LLM)-as-a-Judge scenarios, this experiment introduces a structural bias mechanism into the training data. Under the "biased training" setting, if answer A is in a preceding position, the label is adjusted to indicate that A wins with a certain probability, thus introducing a statistical correlation that "preceding answers are more likely to be judged as superior," as shown in Equations 1-2. The test set does not contain this bias, and the label is always determined solely by quality to evaluate the model's generalization behavior under distribution bias conditions. The specific process is shown in Figure 1, and examples of simulated data are shown in Table 1.

DOI: 10.54254/2753-8818/2026.33416

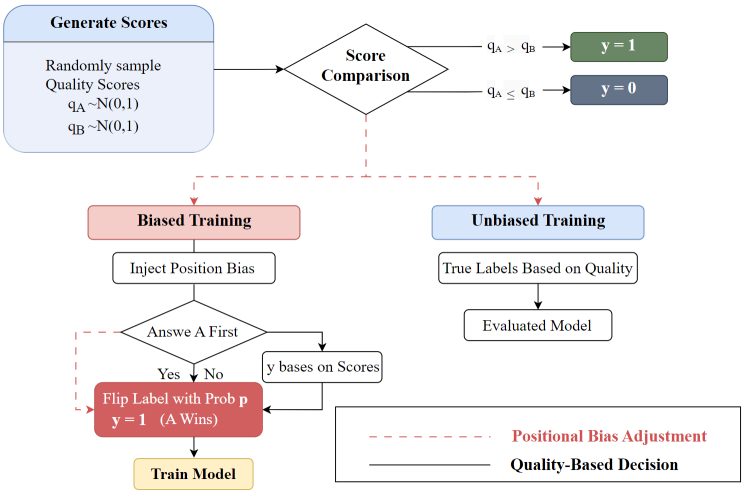


Figure 1. Data generation process

Table 1. Simulated data example

ID	Quality_A	Quality_B	Positon	True_Label	Train_Label
1	0.85	0.42	1	1	1
2	0.13	0.76	1	0	1(flip)
3	0.52	0.11	0	1	1
4	0.22	0.91	1	0	0
5	0.34	0.64	0	0	0

Where Quality_A / Quality_B represent randomly generated answer quality scores, Position indicates whether the answer is in a leading position (value 1 when A is leading), True_Label represents the true label determined solely by quality comparison, and Train_Label is the training label, possibly adjusted from the true label under biased training settings.

Ultimately, this experiment constructed 1000 training samples and 300 test samples. All variables are numerical features, including answer quality and position indicator variables.

3. Method

To analyze whether order relevance in the training data leads to positional bias in the reward model, this study constructs a binary classification reward model. The model input includes the quality features of two answers and their display order indicator variables. The final output is the probability that answer A is better than answer B. The input vector is represented as shown in Equation 3:

$$x = [qA, qB, position] \tag{3}$$

Furthermore, this study uses a logistic regression model as an approximator for the reward function, and its prediction form is shown in Equation 4:

DOI: 10.54254/2753-8818/2026.33416

$$P(A > B) = \sigma(w1qA + w2qB + w3position + b) \tag{4}$$

Where $\sigma(\cdot)$ is the Sigmoid function, w_1, w_2, w_3 are model parameters, and b is the bias term. The model is trained by minimizing the binary cross-entropy loss function, as shown in Equation 5:

$$L = -[ylogP(A > B) + (1 - y)log(1 - P(A > B))] \tag{5}$$

To compare the impact of different training strategies on model bias behavior, this study sets up two training scenarios. For biased training, order correlation is introduced into the training data. When answer A is in a preceding position, it is marked as the winner with a certain probability, thus creating a statistical correlation between positional features and labels. For unbiased training, no order correlation is introduced into the training data; labels are determined entirely by answer quality.

The test set remains unbiased throughout to evaluate the model's generalization ability under uniform distribution conditions. By comparing the magnitude of the positional feature weights w_3 in the model parameters and the position bias rate in the prediction results under the two training methods, this study analyzes whether the model exhibits structural shortcut learning.

4. Results

4.1. Experimental analysis

Table 2. Experimental output results

	Biased Training	Unbiased Training
Accuracy	0.9467	0.9833
Bias Rate	0.5533	0.4833
w_3	1.047	0.303

Table 2 shows that on the test set, the classification accuracy of the Biased Training model is 94.67%, while that of the Unbiased Training model is 98.33%. Both training mechanisms achieve high performance, but Unbiased Training is slightly better than Biased Training.

To evaluate whether the model exhibits order-dependent behavior, the position bias rate was calculated, which is the proportion of times answer A is predicted as the winner when it is in a preceding position. Experimental results show that in the Biased Training case, the model is more inclined to predict the preceding answer as the winner, with a bias rate of 0.553, significantly higher than 0.5; while in the Unbiased Training case, the bias rate is 0.483, close to 0.5.

In the parameters of the logistic regression model, under the Biased Training setting, the position feature weight w_3 = 1.047, which is significantly higher than w_3 = 0.303 in Unbiased Training.

DOI: 10.54254/2753-8818/2026.33416

4.2. Discussion

Empirical evidence indicates that in cases where order and label are statistically correlated in the training data, the model makes use of such a correlation structure to make predictions. Even though the test data are unbiased, the model still has a bias towards positional features. It means that this model is not only able to learn the differences in the quality of answers during training, but also be trained to learn some other structural information of the data distribution.

The accuracy of the model was good in both training conditions. This implies that even though the general discriminative capacity of the model did not decline significantly, the internal decision-making process has altered. Although the general performance indicators are not going down, the discriminative standards of the model can have shifted away, to some degree, the spirit of the task. The phenomenon is a latent threat in real-world systems since with the use of overall measures like accuracy alone it is hard to determine whether structural dependencies have developed within the model.

Reward models are normally trained on preference-based information in large model alignment problems. Provided that the preference data has (1) a fixed order of display (2) variations in source of generation and (3) bias in style consistency during construction, the reward model can learn non-causal properties other than real quality discrimination properties. This bias will also be increased during reinforcement learning stages (e.g. policy optimization with respect to reward models), thereby impacting the distribution of behavior of the resulting generated model.

Thus, despite being performed on a simplified situation, this experiment demonstrates a significant problem: the correlations in the data distribution structure can change how the model makes decisions, but they do not change the short-term performance greatly.

5. Conclusion

This paper deals with the possible problem of positional bias in the LLM-as-a-Judge situation. In terms of data distribution structure, a manageable data environment of synthetic preference was built and the effects of biased and unbiased training mechanisms on the learning behavior of the reward model were compared and analyzed. The findings demonstrate that in biased training conditions the model accuracy is maintained at a reasonably high level (94.67% compared to 98.33% in the control group) yet the weights of the positional features are much higher (1.047 vs 0.303) which means that the model has acquired a structural dependence on the sequential information without a substantial drop in performance.

Further examination shows that such a bias has no substantial impact on the overall predictive accuracy of the model, but alters the discrimination criteria on which the model is based, leading to a mismatch between the actual causal process of the task and an apparent performance stability. This result indicates that when it comes to the evaluation and alignment tasks in the context of LLM, the use of macroscopic measures like accuracy alone cannot be used to assess the credibility of the model; the implicit character of the data distribution can significantly influence the model behavior.

Despite confirming the existence mechanism of positional bias with simple synthetic data and logistic regression model in this study, some limitations are present. To begin with, the experimental setting was somewhat idealized and failed to take into account the semantic complexity and multidimensionality of text generated by real LLM. Second, the model architecture employed was fairly simple and its reliability in both large-scale neural networks or real LLM evaluation models has yet to be confirmed. In addition, this research did not consider other aspects of data distribution like style bias and source bias, but only the single aspect of positional bias.

DOI: 10.54254/2753-8818/2026.33416

Possible future directions of research include: First, to test the generalizability of the conclusions to real LLM-generated data, introduce real data; second, to investigate the propagation and amplification mechanisms of bias in more complex models, combine deep learning models or reinforcement learning frameworks; third, to reduce the reliance of the reward model on non-causal features, test data debiasing and robust training strategies; fourth, build a more robust.

References

[1] He, J., Shi, J., Zhuo, T. Y., Treude, C., Sun, J., Xing, Z., ... & Lo, D. (2026). Llm-as-a-judge for software engineering: Literature review, vision, and the road ahead. ACM Transactions on Software Engineering and Methodology.

[2] Saha, S., Li, X., Ghazvininejad, M., Weston, J., & Wang, T. (2025). Learning to plan & reason for evaluation with thinking-llm-as-a-judge. arXiv preprint arXiv: 2501.18099.

[3] Lambert, N., Pyatkin, V., Morrison, J., Miranda, L. J., Lin, B. Y., Chandu, K., Dziri, N., Kumar, S., Zick, T., Choi, Y., Smith, N. A., & Hajishirzi, H. (2025). RewardBench: Evaluating reward models for language modeling. In Findings of the Association for Computational Linguistics: NAACL (pp. 1755–1797).

[4] Wang, T., Kulikov, I., Golovneva, O., Yu, P., Yuan, W., Dwivedi-Yu, J., ... & Li, X. (2024). Self-taught evaluators. arXiv preprint arXiv: 2408.02666.

[5] Zhang, Q., Wang, Y., Jiang, Y., Li, L., Wu, C., Wang, Y., Jiang, X., Shang, L., Tang, R., Lyu, F., & Ma, C. (2025). Crowd comparative reasoning: Unlocking comprehensive evaluations for LLM-as-a-judge. In Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL) (pp. 5059–5074).

[6] Yu, J., Sun, S., Hu, X., Yan, J., Yu, K., & Li, X. (2025). Improve LLM-as-a-judge ability as a general ability. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP) (pp. 14099–14115).

[7] Whitehouse, C., Wang, T., Yu, P., Li, X., Weston, J., Kulikov, I., & Saha, S. (2025). Incentivizing thinking in LLM-as-a-judge via reinforcement learning. In Proceedings of the 42nd International Conference on Machine Learning (ICML).

[8] Wang, Y., Li, Z., Zang, Y., Wang, C., Lu, Q., Jin, C., & Wang, J. (2025). Unified multimodal chain-of-thought reward model through reinforcement fine-tuning. In Advances in Neural Information Processing Systems (Vol. 38).

[9] Zhou, W., Liu, F., Zheng, H., & Zhao, R. (2025). Mitigating data bias and ensuring reliable evaluation of AI models with shortcut hull learning. Nature Communications, 16(1), 5513.

[10] Li, R., & Gao, Y. (2025). Anchored answers: Unravelling positional bias in GPT-2's multiple-choice questions. In Findings of the Association for Computational Linguistics: ACL (pp. 2439–2465).

[11] Yu, R., Wan, S., Wang, Y., Gao, C. X., Gan, L., Zhang, Z., & Zhan, D. C. (2025). Reward models in deep reinforcement learning: A survey. arXiv preprint arXiv: 2506.15421.