ELSPR: Evaluator LLM Training Data Self-Purification on Non-Transitive Preferences via Tournament Graph Reconstruction

Yan Yu[1], Yilun Liu[2] ✉, Minggui He[2], Shimin Tao[2], Weibin Meng[2], Xinhua Yang[2], Li Zhang[2], Hongxia Ma[2], Dengye Li[3], Daimeng Wei[2], Boxing Chen[4], Fuliang Li[1] ✉

[1]Northeastern University, Shenyang, China
[2]Huawei, Beijing, China
[3]Tongji University, Shanghai, China
[4]Huawei Canada, Montreal, Canada
liuyilun3@huawei.com, lifuliang@cse.neu.edu.cn

Abstract

Pairwise evaluation of large language models (LLMs) has become the dominant paradigm for benchmarking open-ended tasks, yet non-transitive preferences—where evaluators prefer A over B, B over C, but C over A—fundamentally undermine ranking reliability. We show that this critical issue stems largely from low-quality data that contains inherently ambiguous preference pairs. To address this challenge, we propose ELSPR, a principled graph-theoretic framework that models pairwise preferences as tournament graphs and systematically identifies problematic training data. ELSPR quantifies non-transitivity through strongly connected components (SCCs) analysis and measures overall preference clarity using a novel normalized directed graph structural entropy metric. Our filtering methodology selectively removes preference data that induce non-transitivity while preserving transitive preferences. Extensive experiments on the AlpacaEval benchmark demonstrate that models fine-tuned on ELSPR-filtered data achieve substantial improvements: a 13.8% reduction in non-transitivity, a 0.088 decrease in structural entropy, and significantly enhanced discriminative power in real-world evaluation systems. Human validation confirms that discarded data exhibit dramatically lower inter-annotator agreement (34.4% vs. 52.6%) and model-human consistency (51.2% vs. 80.6%) compared to cleaned data. These findings establish ELSPR as an effective data self-purification approach for developing more robust, consistent, and human-aligned LLM evaluation systems.

Code & Datasets — https://github.com/yy0525/ELSPR

1 Introduction

With the rapid advancement of large language model (LLM; OpenAI et al. 2024; Qwen Team 2025; Meta AI 2024) technology, an increasing number of models have become available, making it essential to evaluate their capabilities for selecting the most suitable one. However, existing benchmarks such as MMLU (Hendrycks et al. 2020) and HELM (Liang et al. 2022) have been shown to be insufficient for capturing performance differences in open-ended tasks (Zheng et al. 2024).

✉ Corresponding author.

Given the lack of definitive answers in open-ended tasks, human expert evaluation is considered the gold standard. Yet, due to its high cost and limited scalability, the mainstream approach has shifted toward using LLM-as-a-Judge for efficient evaluation. Recent studies have demonstrated that powerful LLMs, such as GPT-4, can achieve high consistency with human judgments (Zheng et al. 2024; Dubois et al. 2023). Among various approaches, pairwise comparison has emerged as the dominant paradigm, due to its strong alignment with human preferences (Samvelyan et al. 2024; Chen et al. 2024a; Li et al. 2023; Chiang et al. 2024; Liu et al. 2024; Liusie, Manakul, and Gales 2024).

Despite these promising results, LLM-as-a-Judge still suffers from various biases—such as position, verbosity, conformity, selection, and self-reinforcement bias—which compromise evaluation reliability (Xu et al. 2024; Zheng et al. 2024; Koo et al. 2024; Ye et al. 2025; Wei et al. 2024; Choi et al. 2025). A particularly underexplored yet critical issue is the non-transitivity of preferences generated by evaluator LLMs (e.g., $A \succ B$, $B \succ C$, $C \succ A$), where $\succ$ denotes "is preferred to"; that is, $A \succ B$ means A is preferred over B. An illustration is provided in Figure 1. Recent research (Xu et al. 2025) analyzed inconsistencies in the GPT-4 evaluation outcomes when conducting pairwise comparisons between different baseline models. Building on the widely adopted AlpacaEval framework, their findings indicate that preference non-transitivity significantly undermines the robustness of evaluation systems, leading to inconsistent model rankings under various baselines and thereby compromising the reliability of LLM-based evaluators. However, their analysis is limited to observing preference non-transitivity within triplets of samples and does not extend to larger sample sets. Moreover, they do not propose an effective method to substantially reduce the degree of non-transitivity exhibited by evaluator LLMs. These limitations highlight the need for more comprehensive investigations, particularly with larger sample sets, to further explore preference non-transitivity within LLM-based evaluation frameworks.

Worryingly, this issue is also inherited by specialized evaluators such as JudgeLM, PandaLM, and Auto-J (Zhu, Wang, and Wang 2025; Wang et al. 2024b; Li et al. 2024a), which are trained through the distillation of knowledge from

advanced models. The distillation process may inadvertently propagate non-transitive judgment patterns to downstream evaluators.

We hypothesize that the presence of low-quality training data may impair the transitivity of the preferences generated by the evaluator LLM. Many pairwise comparisons, particularly those from open-ended tasks, lack definitive ground truth due to their inherent subjectivity. Human annotators frequently demonstrate significant disagreement, with empirical studies reporting inter-annotator agreement rates as low as 65.7% (Li et al. 2023), indicating that such tasks fundamentally lack universally accepted preference orderings. Consequently, training models with ambiguous and low-quality training data may introduce or exacerbate non-transitive preference relationships, undermining the development of stable and reliable evaluator LLMs. This phenomenon underscores the critical necessity of implementing robust filtering mechanisms to eliminate unreliable data points prior to model training.

In this paper, we present a novel graph-theoretic approach to assess and mitigate preference non-transitivity in evaluator LLMs. Our method, ELSPR, formulates multi-response pairwise comparisons as tournament graphs and systematically filters preference data that induce to overall preference non-transitivity. To quantify preference clarity, we introduce a new metric based on two-dimensional structural entropy of directed graphs. Experimental results demonstrate that models fine-tuned on cleaned data reduce preference non-transitivity by 13.78% and structural entropy by 0.0879, while achieving more robust rankings in real-world evaluation systems, as evidenced by increased standard deviations in MT-bench evaluation framework metrics. Human evaluation studies further validate our approach, confirming that data inducing to overall preference non-transitivity correlates with low inter-annotator agreement among human evaluators and poor alignment between LLM evaluations and human majority votes. These findings underscore ELSPR's effectiveness in developing more reliable evaluation systems for LLMs. Specifically, our contributions are:

- We introduce a graph-theoretic approach to systematically study the non-transitivity problem in preference data generated by evaluator LLM, reveal significant non-transitivity, and propose a two-dimensional structural entropy to quantify preference clarity.
- We propose a robust filtering methodology leveraging tournament graph theory to systematically identify and eliminate preference data that induce non-transitivity in evaluator LLMs. Our experimental results demonstrate that this approach significantly reduces preference non-transitivity, decreases structural entropy, and substantially enhances the evaluation robustness of the resulting models when deployed in real-world evaluation systems.
- We empirically demonstrate through human evaluation that data causing preference non-transitivity is inherently ambiguous and low-quality, with significantly lower inter-annotator agreement (34.4% vs. 52.6%) and lower consistency between evaluator LLM assessments and human majority votes (51.2% vs. 80.6%).

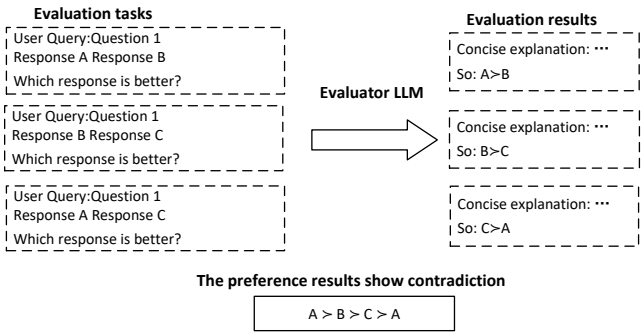


Figure 1: Non-Transitive Preferences in LLM-as-a-Judge for Pairwise Comparisons (e.g., $A \succ B$, $B \succ C$, $C \succ A$).

2 Related Work

2.1 LLM-as-a-Judge and Its Non-Transitive Preferences

The prevailing LLM-as-a-Judge paradigm operates primarily through pairwise comparisons, as evidenced in established frameworks such as VicunaEval, AlpacaEval, and Arena-Hard (Chiang et al. 2023; Li et al. 2023, 2024b). These systems collect responses generated by various LLMs for a given set of questions and then employ advanced LLMs as judges to determine preference orders between response pairs, thus evaluating the relative performance of different models. Recent studies demonstrate that even advanced models such as GPT-4 exhibit significant preference non-transitivity when used in evaluation systems, substantially undermining the reliability of evaluation outcomes. Despite systematic exploration of six distinct prompt templates, improvements in mitigating preference non-transitivity remain limited (Xu et al. 2025). This issue extends to Reinforcement Learning from Human Feedback (RLHF), where non-transitive preferences significantly impair performance. Recent research suggests that this non-transitivity can hinder the learning algorithm from stably converging to the global optimum, leading to preference cycles and thus hindering the effective utilization of preference data (Zhou, Fazel, and Du 2025).

To address this critical challenge, we systematically investigate the underlying conditions that generate non-transitivity in LLM-as-a-Judge systems and introduce a novel methodology to mitigate such inconsistencies.

2.2 Data Selection for LLM Fine-tuning

Previous research (Chen et al. 2024b; Ge et al. 2024; Li et al. 2024c) has established that extracting high-quality subsets from synthetic training sets is crucial for effective fine-tuning. This approach not only enhances the performance of the model, but also substantially reduces the computational requirements. However, this strategy remains underexplored in the domain of LLM-as-a-Judge. Existing methodologies predominantly focus on scaling model architectures or refining engineering techniques, while neglecting the potential benefits of optimizing training data quality.

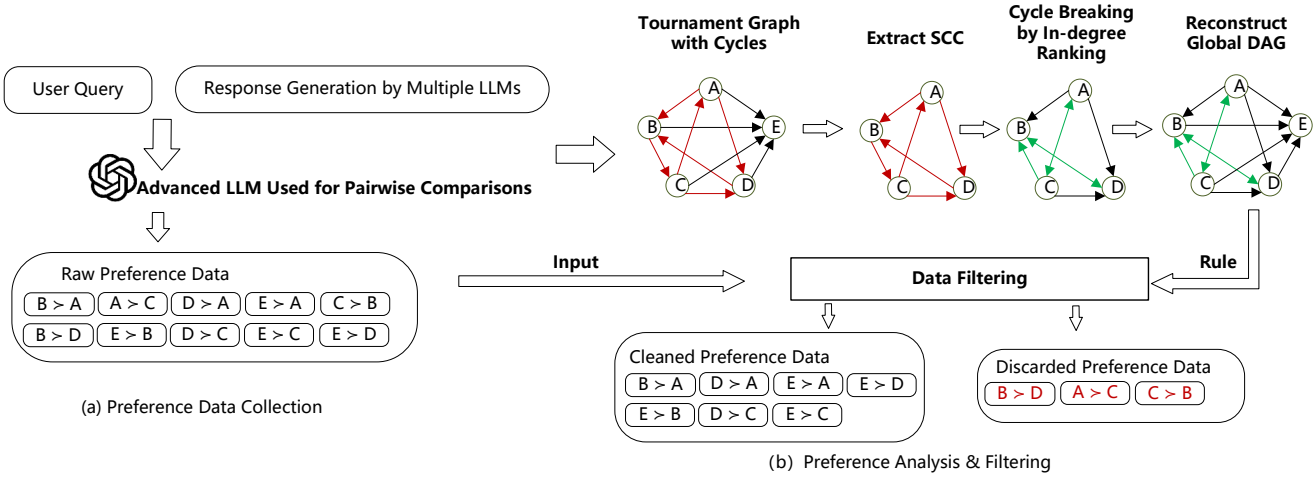


Figure 2: ELSPR (**E**valuator **L**LM training data **S**elf-purification non-transitive **P**references via tournament graph **R**econstruction) framework overview. **(a)** Raw preference data is collected via pairwise comparisons conducted by an advanced LLM. **(b)** The core analysis and filtering process: The raw data is first modeled as a tournament graph to identify cycles within SCCs. These cycles are then broken by reconstructing each SCC into a DAG based on in-degree ranking. The final global DAG serves as a rule to filter the initial raw data, separating it into a cleaned, transitively consistent training set and a discarded set of non-transitive preferences.

3 Methodology

This section presents a graph-theoretic analysis method using tournament graphs to analyze pairwise preferences from evaluator LLMs. The method introduces quality analysis criteria for training data, including non-transitivity detection via strongly connected components (SCCs) and preference clarity analysis via graph entropy. Additionally, a data filtering method is proposed to mitigate non-transitive preferences. Figure 2 outlines the overall framework, with implementation details provided in the following subsections.

3.1 Background

Modeling preferences generated by LLM as a tournament graph. For each question $q_i \in Q$, given the response set $A_i = \{a_1, a_2, ..., a_n\}$ from n LLMs, we construct a tournament graph $G_i = (V_i, E_i)$ through the following procedure: **vertices**: set $V_i = \{v_1, v_2, ..., v_n\}$ corresponds to responses $A_i = \{a_1, a_2, ..., a_n\}$. **Edges**: Defined by preferences generated by evaluator LLM:

$$E_i = \bigcup_{\substack{1 \le j,k \le n \\ j \ne k}} \begin{cases} v_k \to v_j, & \text{if } \mathcal{J}(a_j, a_k) = \text{`win'} \\ & \text{and } \mathcal{J}(a_k, a_j) = \text{`lose'} \\ v_j \to v_k, & \text{if } \mathcal{J}(a_j, a_k) = \text{`lose'} \\ & \text{and } \mathcal{J}(a_k, a_j) = \text{`win'} \\ v_j \leftrightarrow v_k, & \text{otherwise} \end{cases} \quad (1)$$

Here, $\mathcal{J}(a_j, a_k) \in \{\text{`win'}, \text{`lose'}\}$ denotes the pairwise comparison result between answers a_j and a_k, where a_j appears before a_k in the prompt.

Considering the common position bias in evaluator LLMs (Wang et al. 2024a), we apply position swapping by comparing each response pair in both orders: $\mathcal{J}(a_j, a_k)$ and $\mathcal{J}(a_k, a_j)$. This ensures a more robust and balanced assessment (Zheng et al. 2024). If the preferences differ across orders—indicating possible position bias—a bidirectional edge is added between the corresponding vertices to represent a 'tie'.

3.2 Quality Analysis Framework for Evaluator LLM Training Data

We introduce a framework for analyzing evaluator LLM training data quality. Leveraging directed graph representations of preference tournaments, the framework introduces two metrics: SCC analysis quantifying preference non-transitivity and two-dimensional structural entropy measuring preference clarity.

Quantifying preference non-transitivity via SCC analysis. SCCs are maximal subgraphs where any two vertices are mutually reachable through directed paths. This property provides a natural mechanism for identifying preference cycles—a direct manifestation of non-transitivity in evaluator LLM judgments. When vertex v_i can reach vertex v_j via a directed path, this indicates a preference chain $a_j \succ \cdot \succ a_i$. The simultaneous existence of paths from v_i to v_j and from v_j to v_i represents contradictory preference relationships, signaling a violation of transitivity.

We employ Tarjan's algorithm (Tarjan 1972) to efficiently identify SCCs in our directed graphs. Tarjan's algorithm operates in linear time, specifically $\mathcal{O}(|V| + |E|)$, where $|V|$ and $|E|$ denote the number of vertices and edges, respectively, ensuring scalability to large graphs. Since non-transitivity requires at least three elements in a cycle, we focus on SCCs containing more than two vertices. Additionally, we exclude cases where every vertex pair shares bidirectional edges (indicating 'tie'), as these represent consistent preference relations. For example, the pattern $(A = B)$, $(B = C)$, $(C = A)$ constitutes a valid SCC but maintains

transitive preference relationships. Formally, we identify the set of non-transitive SCCs as:

$$\mathcal{S}_{\text{n-t}} = \left\{ S \in \text{SCCs}(G) \;\middle|\; \begin{array}{r} |S| > 2 \wedge \exists v_j, v_k \in S, \\ (v_j \leftrightarrow v_k) \notin E(S) \end{array} \right\}, \quad (2)$$

where $\text{SCCs}(G)$ represents all SCC in graph G, $|S|$ denotes the component size, and $(v_j \leftrightarrow v_k) \notin E(S)$ indicates the absence of bidirectional edges between some vertices.

To quantify the prevalence of non-transitivity across our entire training set, we compute the non-transitivity ratio:

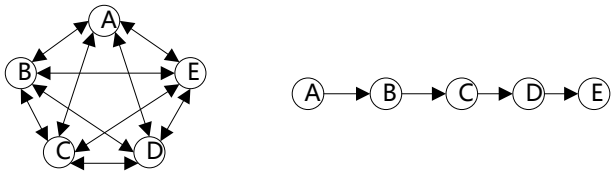
$$\rho_{\text{non-trans}} = \frac{\sum_{q_i \in \mathbf{Q}} |S_{\text{n-t}}(G_i)|}{\sum_{q_i \in \mathbf{Q}} |V_i|}, \quad (3)$$

where the numerator represents the total number of vertices in non-transitive SCCs across all questions $\mathbf{Q}$, and the denominator represents the total number of vertices across all graphs. This metric ranges from 0 to 1, with higher values indicating greater prevalence of non-transitive in the evaluator LLM's preferences. A perfectly transitive preference system would yield $\rho_{\text{non-trans}} = 0$, while completely cyclical preferences would approach $\rho_{\text{non-trans}} = 1$.

Analysis of preference clarity based the structural entropy of directed graph. While the SCC-based approach effectively quantifies preference non-transitivity, this metric alone is insufficient to characterize preference linearity. For instance, a dataset comprised entirely of preference 'tie' would exhibit perfect transitivity yet fail to establish any linear ordering. To address this limitation, we introduce directed graph entropy as a complementary measurement. Structural entropy (Li and Pan 2016) extends Shannon entropy (Shannon 1948) to directed graphs, providing a measure of system uncertainty and the complexity of relationships within the graph. These concepts have been widely applied across various domains (Zou et al. 2024; Duan et al. 2024; Peng et al. 2024; Hou et al. 2025). We introduce and refine the two-dimensional structural entropy metric for directed graphs to analyze the overall clarity of preferences in evaluator LLMs.

A crucial observation drives our methodology is that SCCs composed of single vertices inherently exhibit strict transitivity. Preference relations among such singleton components naturally form a linear order that does not increase preference complexity. For example, consider preferences $A \succ B$, $B \succ C$, and $A \succ C$. These form a clear linear order $A \succ B \succ C$, where each vertex constitutes its own singleton SCC. This clarity reflects the inherent transitivity of singleton SCCs. Based on this insight, we adopt SCCs as the basic community units when calculating structural entropy. To ensure accurate quantification of meaningful structural complexity, we exclude interactions between pure single-point SCCs. Instead, we retain only interactions between singleton SCCs and multi-vertex SCCs, as well as interactions between different multi-vertex SCCs. This selective approach ensures more accurate quantification of meaningful structural complexity. Consequently, preference relations closer to a linear order produce lower entropy values.

Figure 3 illustrates two contrasting scenarios that provide an intuitive explanation of the two-dimensional structural entropy in tournament graph preference relations. Figure 3 (a) depicts a high entropy case with a chaotic tournament containing multiple preference cycles resulting in complex non-transitive relationships. In contrast, Figure 3 (b) shows a low entropy case with a perfectly ordered tournament where responses form a clear linear hierarchy, producing a simple transitive structure.



(a) High Entropy Case (b) Low Entropy Case

Figure 3: Cases of High and Low Structural Entropy in Preference Tournaments.

Given a directed graph $G = (V, E)$ with $n = |V|$ vertices, we perform SCC decomposition resulting in $\text{SCCs}(G) = \{SCC_1, SCC_2, \ldots, SCC_L\}$, where L is the total number of SCCs. For any vertex $v \in V$, we denote $d_{\text{in}}(v)$ and $d_{\text{out}}(v)$ as the in-degree and out-degree of v in G. For any SCC_i, we define its volume as $v(SCC_i) = \sum_{v \in SCC_i} d_{\text{in}}(v)$, while $v(G)$ represents the total in-degree of the entire graph.

The two-dimensional structural entropy is computed as:

$$\begin{aligned} H^2(G) = &- \sum_{j=1}^{L} \frac{g_j}{v(G)} \log_2 \frac{v(SCC_j)}{v(G)} \\ &- \sum_{j=1}^{L} \frac{v(SCC_j)}{v(G)} \left(\sum_{v \in SCC_j} \frac{d_{\text{in}}(v)}{v(SCC_j)} \log_2 \frac{d_{\text{in}}(v)}{v(SCC_j)} \right), \end{aligned} \quad (4)$$

The first term captures the entropy of the partition (inter-community complexity), while the second term captures the weighted average of entropies within each SCC (intra-community complexity). The variable g_j represents the number of incoming edges to SCC_j from vertices outside this component, specifically counting edges between singleton SCCs and multi-vertex SCCs, as well as edges between different multi-vertex SCCs. This value quantifies the external influence on each SCC and plays a crucial role in measuring cross-community complexity.

To facilitate meaningful comparisons across graphs of different sizes, we normalize the structural entropy. The normalized structural entropy for a graph G is defined as:

$$\tau(G) = \frac{H^2(G)}{\log_2 n}. \quad (5)$$

Normalization is necessary since raw entropy increases with graph size. Dividing by $\log_2 n$ scales the entropy to a range between 0 and 1, where values closer to 0 indicate more ordered structures approaching linear hierarchy, while values closer to 1 suggest higher complexity and less clear preference ordering. The upper bound $\log_2 n$ represents the

maximum possible entropy for a graph with n vertices, corresponding to a completely uniform distribution of structural information. To evaluate the overall preference clarity of an evaluator LLM across a set of questions $\mathbf{Q}$, we compute the average normalized structural entropy:

$$\tau_{\text{avg}} = \frac{\sum_{q_i \in \mathbf{Q}} \tau(G_i)}{|\mathbf{Q}|}, \tag{6}$$

Here, $\tau(G_i)$ is the normalized structural entropy for the graph G_i corresponding to question q_i, and $|\mathbf{Q}|$ is the total number of questions in the set $\mathbf{Q}$.

The τ_{avg} metric quantifies preference clarity: higher values indicate complex, non-transitive patterns, while lower values reflect relationships closer to linear order. It enables objective assessment of consistency across evaluator LLMs, where lower entropy denotes more coherent and interpretable preferences.

3.3 Filtering Strategy for Preference Data That Induce Non-Transitivity

If cycles within each SCC of a directed graph can be eliminated to form DAGs, the entire graph can be transformed into a DAG. Our framework leverages this property by converting each SCC into a DAG while preserving inter-SCC preference relations, ensuring global acyclicity without altering component relationships. From the resulting DAG, we derive transitive preference relations to filter evaluator LLMs' preference data, enabling training data self-purification. The procedure is:

1. For each SCC_i compute the in-degree e_i^{in} for each vertex v_i, representing its global 'win' score. Vertices with higher in-degree scores are prioritized over those with lower scores. To reconstruct the internal edges within SCC_i, first remove all original edges between vertices within SCC_i. Then, for any pair v_i, v_j: if $e_i^{\text{in}} > e_j^{\text{in}}$, add a directed edge $v_j \to v_i$; if $e_i^{\text{in}} = e_j^{\text{in}}$, add a bidirectional edge $v_i \leftrightarrow v_j$.
2. After processing all SCCs, the original cyclic graph is transformed into a DAG. This DAG is used to filter the training data as follows: For bidirectional edges $(v_i \leftrightarrow v_j)$, the correct preferences are recorded as $\mathcal{J}(a_i, a_j) =$ 'tie' and $\mathcal{J}(a_j, a_i) =$ 'tie'. For unidirectional edges $(v_i \to v_j)$, the correct preferences are recorded as $\mathcal{J}(a_i, a_j) =$ 'lose' and $\mathcal{J}(a_j, a_i) =$ 'win'.
3. The training data are traversed sequentially. All instances that are consistent with the correct preferences are added to the training set "Cleaned", while the remaining data are placed in the training set "Discarded".

The final "Cleaned" training data only retains linearly transitive preference data, resulting in $\rho_{\text{non-trans}} = 0$ and $\tau_{\text{avg}} = 0$. Our approach balances optimality and scalability, with an overall time complexity of $\mathcal{O}(|V|^2 + |E|)$, consisting of $\mathcal{O}(|V|^2)$ for graph construction, $\mathcal{O}(|V| + |E|)$ for SCC decomposition, $\mathcal{O}(|V|^2)$ for intra-SCC reconstruction, and $\mathcal{O}(|E|)$ for edge filtering. The detailed algorithmic procedure is provided in Appendix A[1].

4 Experiment Setup

4.1 Dataset

In this study, we conduct experimental validation using the AlpacaEval benchmark (Li et al. 2023). AlpacaEval is specifically designed to assess the overall capabilities of LLMs in open-ended tasks, covering a wide range of evaluation scenarios such as reasoning and text and code generation. The benchmark comprises five datasets, Helpful_Base, Oasst, Koala, Vicuna, and Self-Instruct.

4.2 Preference Data Collection

We evaluated Qwen2.5-Max (Qwen Team 2025) on 2.5k human-annotated samples from AlpacaEval using the chain-of-thought (COT) comparison (Li et al. 2023). Qwen2.5-Max achieved a human agreement rate of 68.9%. Based on these results, we selected it as our teacher model for preference data generation.

4.3 Experiment Details

21 representative LLMs were selected from the AlpacaEval leaderboard (14 for training, 7 for testing). All experiments employed CoT comparison templates with temperature set to 0, and preferences were generated via the Qwen2.5-Max API. The unfiltered "Raw" datasets were processed according to the procedure described in Section 3.3 to produce "Cleaned" training sets; the corresponding distributions are illustrated in Figure 4. Qwen2.5-7B-Instruct was fine-tuned on both versions of each of the five datasets using LoRA (Hu et al. 2022) (rank $= 8$, 3 epochs, learning rate $= 1 \times 10^{-4}$, batch size $= 16$).

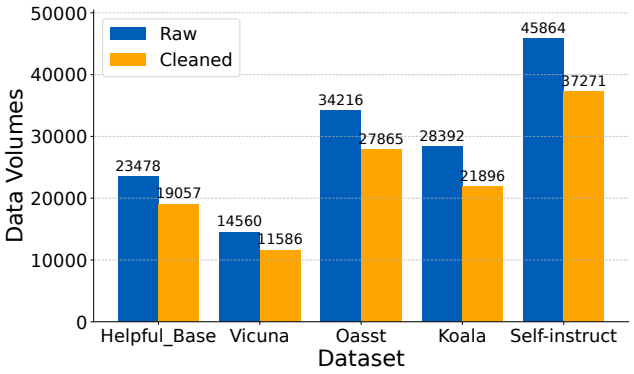


Figure 4: Comparison of data volumes between **"Raw"** and **"Cleaned"** training sets across datasets. The **"Cleaned"** training set's volume is approximately 80% of the **"Raw"** training set for each dataset.

[1]Due to page limits, appendices are available only in the arXiv version (Yu et al. 2025).

Model	Helpful_Base		Vicuna		Oasst		Koala		Self-instruct	
	$\rho^{\downarrow}_{\text{non-trans}}$	$\tau^{\downarrow}_{\text{avg}}$	$\rho^{\downarrow}_{\text{non-trans}}$	$\tau^{\downarrow}_{\text{avg}}$	$\rho^{\downarrow}_{\text{non-trans}}$	$\tau^{\downarrow}_{\text{avg}}$	$\rho^{\downarrow}_{\text{non-trans}}$	$\tau^{\downarrow}_{\text{avg}}$	$\rho^{\downarrow}_{\text{non-trans}}$	$\tau^{\downarrow}_{\text{avg}}$
Stronger LLMs										
Qwen2.5-Max	63.7%	0.805	75.4%	0.845	64.3%	0.788	71.5%	0.830	65.0%	0.780
Base models and Variants										
Qwen-Base	82.8%	0.922	78.9%	0.891	83.4%	0.914	81.0%	0.910	81.5%	0.912
Qwen-Raw	62.0%	0.796	57.5%	0.803	55.8%	0.773	64.3%	0.816	59.7%	0.767
Qwen-Random	60.6%	0.790	63.6%	0.808	58.8%	0.771	60.4%	0.812	57.1%	0.764
Qwen-Cleaned (ours)	**44.9%**	**0.700**	**43.9%**	**0.726**	**47.0%**	**0.694**	**48.5%**	**0.715**	**49.0%**	**0.680**
LLaMA-Base	76.4%	0.852	67.0%	0.846	67.4%	0.793	69.9%	0.818	71.0%	0.809
LLaMA-Raw	59.0%	0.765	60.2%	0.772	58.2%	0.760	57.2%	0.746	60.0%	0.783
LLaMA-Cleaned (ours)	**40.2%**	**0.652**	**45.4%**	**0.691**	**43.0%**	**0.629**	**44.4%**	**0.659**	**42.0%**	**0.642**

Table 1: Comparison of **Preference Non-Transitivity** and **Overall Clarity** for evaluator LLMs. Qwen-Base denotes to the original Qwen2.5-7B-Instruct model, while Qwen-Raw, Qwen-Random, and Qwen-Cleaned denote models fine-tuned on the **"Raw"**, **"Random"**, and **"Cleaned"** training sets, respectively. For example, Qwen-Cleaned in the Helpful_Base column reflects the performance of the model fine-tuned on the "Cleaned" training set derived from filtered Helpful_Base training set.

5 Results and Analysis

This section verifies the effectiveness of the proposed filtering method on five datasets. The analysis begins with an examination of preference non-transitivity degree and overall preference clarity of evaluator LLMs. To further demonstrate the method's effectiveness, experiments based on actual usage scenarios are conducted using the MT-bench evaluation framework to calculate the standard deviation between final results. Additionally, a detailed data quality analysis examines two key aspects of manual evaluation: (1) consistency between manual annotators, and (2) consistency between evaluation results and human majority vote.

5.1 Main Results

Analysis of preference non-transitivity and clarity. As shown in Table 1, models fine-tuned on the "Cleaned" training set demonstrate the lowest preference non-transitivity and highest preference clarity across all datasets, including Qwen2.5-Max, highlighting the effectiveness of our proposed data filtering methodology. To verify that models fine-tuned on the "Cleaned" training set achieve lower preference non-transitivity and higher preference clarity even on "unseen" questions, we conducted cross-validation as our out-of-domain experiment by testing each fine-tuned model across five test sets. As demonstrated in Table 2, models fine-tuned on "Cleaned" training set exhibited an average reduction in non-transitivity of 13.8% compared to those trained on "Raw" training set. Table 3 shows a reduction of 0.088 in normalized structural entropy, indicating clearer overall preferences. Detailed results are in Appendix B.

Human validation of discarded data quality. To verify the low quality of discarded data, we evaluated 100 randomly sampled instances from both "Cleaned" and "Discarded" sets across five datasets (1,000 total instances). Three independent annotators assessed these samples, enabling measurement of inter-annotator consistency and model-human alignment. Results in Table 4 indicate that human annotator consistency in the "Discard" training set averages only 34.4%, significantly lower than the 52.6% observed in the "Cleaned" training set. This confirms that data leading to non-transitive preferences is inherently ambiguous. Furthermore, model evaluation reveals that consistency between the "Discarded" training set and human majority votes averages merely 51.2%, substantially below the 80.6% achieved by the "Cleaned" training set. These findings provide compelling evidence that the "Discarded" training set contains low-quality preference pairs unsuitable for training models to learn consensus human preferences.

For deeper insight into non-transitive preferences, we observed that source response pairs exhibiting non-transitivity demonstrate higher textual similarity. This observation suggests that preference cycles are likely to occur when the

Dataset	Raw	Cleaned	Δ
Helpful_Base	66.0%	50.3%	-15.7%
Vicuna	61.6%	51.0%	-10.6%
Oasst	62.6%	48.7%	-13.9%
Koala	64.1%	48.7%	-15.4%
Self-instruct	67.2%	53.8%	-13.4%
Average	64.3%	50.5%	-13.8%

Table 2: Comparison of $\rho_{\text{non-trans}}$ (**average preference non-transitivity**) between models fine-tuned on **"Raw"** and **"Cleaned"** training sets using Qwen2.5-7B-Instruct.

Dataset	Raw	Cleaned	Δ
Helpful_Base	0.820	0.717	-0.103
Vicuna	0.806	0.737	-0.069
Oasst	0.797	0.710	-0.087
Koala	0.804	0.718	-0.086
Self-instruct	0.829	0.735	-0.094
Average	0.811	0.723	-0.088

Table 3: Comparison of τ_{avg} (**average preference clarity**) between models fine-tuned on **"Raw"** and **"Cleaned"** training sets using Qwen2.5-7B-Instruct.

	HC		MHA	
Dataset	**Cleaned**	**Discarded**	**Cleaned**	**Discarded**
Help.	59%	46%	72%	45%
Vicuna	48%	24%	78%	47%
Oasst	56%	33%	83%	53%
Koala	54%	41%	84%	61%
Self.	46%	28%	86%	50%
Average	52.6%	34.4%	80.6%	51.2%

Table 4: Comparison of Human Evaluation Consistency and Model-Human Agreement between **training sets**. HC: Human Consistency (inter-annotator consensus); MHA: Model-Human Agreement (alignment with majority vote). Help. = Helpful_Base; Self. = Self-instruct.

Pref.	**Helpful.**	**Vicuna**	**Oasst**	**Koala**	**Self.**
Non.	0.0903	0.1026	0.0946	0.0888	0.1135
Trans.	0.0713	0.0918	0.0788	0.0746	0.0829

Table 5: Comparison of **source text pair** Self-Bleu scores across datasets. Non. and Trans. indicate source pairs from non-transitive and transitive preferences; Helpful. = Helpful_Base; Self. = Self-instruct.

quality difference between source responses falls below the just noticeable difference (JND) (Stern and Johnson 2010) threshold of the evaluator LLM. As shown in Table 5, we quantified content similarity between response pairs by calculating Self-Bleu scores. Source response pairs with non-transitive preference relations consistently displayed higher Self-Bleu values than those with transitive relations, confirming that preference non-transitivity correlates with greater text similarity between comparative samples.

Performance analysis of real-world evaluation systems. We compared the performance of our trained evaluator LLMs within the MT-bench evaluation framework (Zheng et al. 2024). We adopted the adjusted win rate calculation consistent with the MT-bench evaluation methodology. This metric normalizes performance by treating ties as partial wins, defined as $r_{\text{adj}} = (r_w + 0.5 \cdot r_t)/(r_w + r_l + r_t)$, where r_w, r_l, and r_t represent the win rate, loss rate, and tie rate respectively. The adjusted rate r_{adj} serves as the primary indicator for our model ranking. Results in Table 6 demonstrate that models fine-tuned on the "Cleaned" training sets exhibit significantly higher standard deviations in their adjusted win rates across five datasets. This indicates enhanced robustness of the evaluation results and better differentiation of performance differences between models.

Dataset	**Help.**	**Vicu.**	**Oasst**	**Koala**	**Self.**
Cleaned	16.1%	16.6%	13.1%	14.9%	9.7%
Raw	12.2%	13.6%	11.0%	12.3%	8.2%
Δ	+3.9%	+3.0%	+2.1%	+2.6%	+1.5%

Table 6: Standard deviation analysis. Help. = Helpful_Base; Vicu. = Vicuna; Self. = Self-instruct.

	MHA		SC	
Dataset	**Raw**	**Cleaned**	**Raw**	**Cleaned**
Helpful_base	66.9%	66.9%	0.93	0.97
Vicuna	65.2%	65.8%	0.97	0.98
Oasst	66.4%	67.6%	0.93	0.93
Koala	66.4%	66.9%	0.98	0.98
Self-instruct	67.8%	68.3%	0.98	1.00
Average	66.5%	67.1%	0.96	0.97

Table 7: Comparison of Model-Human Agreement (MHA) and Spearman Correlation (SC) between **fine-tuned models**. Spearman Correlation quantifies the rank correlation between model and human preferences.

Analysis of human agreement impact. The model's alignment with human preferences was evaluated by calculating agreement rates and Spearman correlation coefficients on 2.5k manually annotated preference labels from AlpacaEval. Results in Table 7 indicate that models fine-tuned on the "Cleaned" training set consistently outperformed those trained on "Raw" training set across multiple dataset, with maximum improvements of 1.2% in human agreement and 0.04 in Spearman correlation. These results demonstrate that filtering ambiguous, low-quality preference data not only improves discrimination and enhances the performance of downstream evaluation systems but also increases consistency with human judgment, providing strong evidence for practical applications.

5.2 Ablation Studies

Effect of Different Base Models: Testing using the same steps with LLaMA3.1-8B-Instruct confirms our findings: models trained on the "Cleaned" training set consistently outperformed those trained on the "Raw" training set (Table 1). **Data Filtering Analysis:** Our "Cleaned" set retained approximately 80% of the raw data (Figure 4). A control experiment with random 20% filtering (Qwen-Random) produced results similar to training on the "Raw" set but with higher preference non-transitivity, confirming that our targeted approach improves data quality beyond simple reduction (Table 1). **Prompt Format Variations:** Additional experiments with prompts explicitly allowing "tie" judgments further validated our methodology. Detailed prompt templates and results are provided in Appendices B and C.

6 Conclusion

We propose ELSPR, a graph-theoretic framework to analyze and mitigate non-transitivity in evaluator LLMs by modeling pairwise comparisons as a tournament graph and filtering problematic data. Ambiguous, low-quality preferences are a major source of non-transitive judgments, and targeted filtering improves evaluation reliability. Empirical results show that filtered data aligns better with human judgments and can fine-tune models to reduce non-transitive bias and structural entropy. This work underscores the importance of data quality and provides a scalable approach to enhance consistency on future benchmarks.

Acknowledgments

This work is supported by the National Natural Science Foundation of China under Grant Nos. 62572105 and U22B2005; the Liaoning Revitalization Talents Program under Grant No. XLYC2403086.

References

Chen, G. H.; Chen, S.; Liu, Z.; Jiang, F.; and Wang, B. 2024a. Humans or LLMs as the Judge? A Study on Judgement Bias. In Al-Onaizan, Y.; Bansal, M.; and Chen, Y.-N., eds., *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 8301–8327. Miami, Florida, USA: Association for Computational Linguistics.

Chen, L.; Li, S.; Yan, J.; Wang, H.; Gunaratna, K.; Yadav, V.; Tang, Z.; Srinivasan, V.; Zhou, T.; Huang, H.; and Jin, H. 2024b. AlpaGasus: Training a Better Alpaca with Fewer Data. In *The Twelfth International Conference on Learning Representations*.

Chiang, W.-L.; Li, Z.; Lin, Z.; Sheng, Y.; Wu, Z.; Zhang, H.; Zheng, L.; Zhuang, S.; Zhuang, Y.; Gonzalez, J. E.; Stoica, I.; and Xing, E. P. 2023. Vicuna: An Open-Source Chatbot Impressing GPT-4 with 90%* ChatGPT Quality.

Chiang, W.-L.; Zheng, L.; Sheng, Y.; Angelopoulos, A. N.; Li, T.; Li, D.; Zhu, B.; Zhang, H.; Jordan, M.; Gonzalez, J. E.; and Stoica, I. 2024. Chatbot Arena: An Open Platform for Evaluating LLMs by Human Preference. In *Forty-first International Conference on Machine Learning*.

Choi, H. K.; Xu, W.; Xue, C.; Eckman, S.; and Reddy, C. K. 2025. Mitigating Selection Bias with Node Pruning and Auxiliary Options. In Che, W.; Nabende, J.; Shutova, E.; and Pilehvar, M. T., eds., *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 5190–5215. Vienna, Austria: Association for Computational Linguistics. ISBN 979-8-89176-251-0.

Duan, L.; Chen, X.; Liu, W.; Liu, D.; Yue, K.; and Li, A. 2024. Structural Entropy Based Graph Structure Learning for Node Classification. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(8): 8372–8379.

Dubois, Y.; Li, C. X.; Taori, R.; Zhang, T.; Gulrajani, I.; Ba, J.; Guestrin, C.; Liang, P. S.; and Hashimoto, T. B. 2023. AlpacaFarm: A Simulation Framework for Methods that Learn from Human Feedback. In Oh, A.; Naumann, T.; Globerson, A.; Saenko, K.; Hardt, M.; and Levine, S., eds., *Advances in Neural Information Processing Systems*, volume 36, 30039–30069. Curran Associates, Inc.

Ge, Y.; Liu, Y.; Hu, C.; Meng, W.; Tao, S.; Zhao, X.; Xia, M.; Li, Z.; Chen, B.; Yang, H.; Li, B.; Xiao, T.; and Zhu, J. 2024. Clustering and Ranking: Diversity-preserved Instruction Selection through Expert-aligned Quality Estimation. In Al-Onaizan, Y.; Bansal, M.; and Chen, Y.-N., eds., *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 464–478. Miami, Florida, USA: Association for Computational Linguistics.

Hendrycks, D.; Burns, C.; Basart, S.; Zou, A.; Mazeika, M.; Song, D.; and Steinhardt, J. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

Hou, Y.; Zhu, H.; Liu, R.; Su, Y.; Xia, J.; Wu, J.; and Xu, K. 2025. Structural Entropy Guided Unsupervised Graph Out-Of-Distribution Detection. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(16): 17258–17266.

Hu, E. J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; and Chen, W. 2022. LoRA: Low-Rank Adaptation of Large Language Models. In *International Conference on Learning Representations*.

Koo, R.; Lee, M.; Raheja, V.; Park, J. I.; Kim, Z. M.; and Kang, D. 2024. Benchmarking Cognitive Biases in Large Language Models as Evaluators. In Ku, L.-W.; Martins, A.; and Srikumar, V., eds., *Findings of the Association for Computational Linguistics: ACL 2024*, 517–545. Bangkok, Thailand: Association for Computational Linguistics.

Li, A.; and Pan, Y. 2016. Structural Information and Dynamical Complexity of Networks. *IEEE Transactions on Information Theory*, 62(6): 3290–3339.

Li, J.; Sun, S.; Yuan, W.; Fan, R.-Z.; hai zhao; and Liu, P. 2024a. Generative Judge for Evaluating Alignment. In *The Twelfth International Conference on Learning Representations*.

Li, T.; Chiang, W.-L.; Frick, E.; Dunlap, L.; Wu, T.; Zhu, B.; Gonzalez, J. E.; and Stoica, I. 2024b. From Crowdsourced Data to High-Quality Benchmarks: Arena-Hard and Bench-Builder Pipeline. *CoRR*, abs/2406.11939.

Li, X.; Zhang, T.; Dubois, Y.; Taori, R.; Gulrajani, I.; Guestrin, C.; Liang, P.; and Hashimoto, T. B. 2023. AlpacaEval: An Automatic Evaluator of Instruction-following Models. https://github.com/tatsu-lab/alpaca_eval.

Li, Y.; Hui, B.; Xia, X.; Yang, J.; Yang, M.; Zhang, L.; Si, S.; Chen, L.-H.; Liu, J.; Liu, T.; Huang, F.; and Li, Y. 2024c. One-Shot Learning as Instruction Data Prospector for Large Language Models. In Ku, L.-W.; Martins, A.; and Srikumar, V., eds., *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 4586–4601. Bangkok, Thailand: Association for Computational Linguistics.

Liang, P.; Bommasani, R.; Lee, T.; Tsipras, D.; Soylu, D.; Yasunaga, M.; Zhang, Y.; Narayanan, D.; Wu, Y.; Kumar, A.; et al. 2022. Holistic evaluation of language models. *arXiv preprint arXiv:2211.09110*.

Liu, Y.; Zhou, H.; Guo, Z.; Shareghi, E.; Vulic, I.; Korhonen, A.; and Collier, N. 2024. Aligning with Human Judgement: The Role of Pairwise Preference in Large Language Model Evaluators. *arXiv preprint arXiv:2403.16950*.

Liusie, A.; Manakul, P.; and Gales, M. 2024. LLM Comparative Assessment: Zero-shot NLG Evaluation through Pairwise Comparisons using Large Language Models. In Graham, Y.; and Purver, M., eds., *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, 139–151. St. Julian's, Malta: Association for Computational Linguistics.

Meta AI. 2024. Introducing Llama 3.1: Our Most Capable Models to Date. https://ai.meta.com/blog/meta-llama-3-1/.

OpenAI; Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F. L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; and othres. 2024. GPT-4 Technical Report. arXiv:2303.08774.

Peng, H.; Zhang, J.; Huang, X.; Hao, Z.; Li, A.; Yu, Z.; and Yu, P. S. 2024. Unsupervised Social Bot Detection via Structural Information Theory. *ACM Trans. Inf. Syst.*, 42(6).

Qwen Team. 2025. Qwen2.5-Max: Exploring the Intelligence of Large-Scale MoE Models. *Qwen Blog*. Accessed on 2025-04-05.

Samvelyan, M.; Raparthy, S. C.; Lupu, A.; Hambro, E.; Markosyan, A. H.; Bhatt, M.; Mao, Y.; Jiang, M.; Parker-Holder, J.; Foerster, J.; Rocktäschel, T.; and Raileanu, R. 2024. Rainbow Teaming: Open-Ended Generation of Diverse Adversarial Prompts. In Globerson, A.; Mackey, L.; Belgrave, D.; Fan, A.; Paquet, U.; Tomczak, J.; and Zhang, C., eds., *Advances in Neural Information Processing Systems*, volume 37, 69747–69786. Curran Associates, Inc.

Shannon, C. E. 1948. A mathematical theory of communication. *The Bell System Technical Journal*, 27(3): 379–423.

Stern, M. K.; and Johnson, J. H. 2010. Just noticeable difference. *The corsini encyclopedia of psychology*, 1–2.

Tarjan, R. 1972. Depth-First Search and Linear Graph Algorithms. *SIAM Journal on Computing*, 1(2): 146–160.

Wang, P.; Li, L.; Chen, L.; Cai, Z.; Zhu, D.; Lin, B.; Cao, Y.; Kong, L.; Liu, Q.; Liu, T.; and Sui, Z. 2024a. Large Language Models are not Fair Evaluators. In Ku, L.-W.; Martins, A.; and Srikumar, V., eds., *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 9440–9450. Bangkok, Thailand: Association for Computational Linguistics.

Wang, Y.; Yu, Z.; Yao, W.; Zeng, Z.; Yang, L.; Wang, C.; Chen, H.; Jiang, C.; Xie, R.; Wang, J.; Xie, X.; Ye, W.; Zhang, S.; and Zhang, Y. 2024b. PandaLM: An Automatic Evaluation Benchmark for LLM Instruction Tuning Optimization. In *The Twelfth International Conference on Learning Representations*.

Wei, S.-L.; Wu, C.-K.; Huang, H.-H.; and Chen, H.-H. 2024. Unveiling Selection Biases: Exploring Order and Token Sensitivity in Large Language Models. In Ku, L.-W.; Martins, A.; and Srikumar, V., eds., *Findings of the Association for Computational Linguistics: ACL 2024*, 5598–5621. Bangkok, Thailand: Association for Computational Linguistics.

Xu, W.; Zhu, G.; Zhao, X.; Pan, L.; Li, L.; and Wang, W. 2024. Pride and Prejudice: LLM Amplifies Self-Bias in Self-Refinement. In Ku, L.-W.; Martins, A.; and Srikumar, V., eds., *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 15474–15492. Bangkok, Thailand: Association for Computational Linguistics.

Xu, Y.; Ruis, L.; Rocktäschel, T.; and Kirk, R. 2025. Investigating Non-Transitivity in LLM-as-a-Judge. *arXiv preprint arXiv:2502.14074*.

Ye, J.; Wang, Y.; Huang, Y.; Chen, D.; Zhang, Q.; Moniz, N.; Gao, T.; Geyer, W.; Huang, C.; Chen, P.-Y.; Chawla, N. V.; and Zhang, X. 2025. Justice or Prejudice? Quantifying Biases in LLM-as-a-Judge. In *The Thirteenth International Conference on Learning Representations*.

Yu, Y.; Liu, Y.; He, M.; Tao, S.; Meng, W.; Yang, X.; Zhang, L.; Ma, H.; Li, D.; Wei, D.; Chen, B.; and Li, F. 2025. ELSPR: Evaluator LLM Training Data Self-Purification on Non-Transitive Preferences via Tournament Graph Reconstruction. arXiv:2505.17691.

Zheng, L.; Chiang, W.-L.; Sheng, Y.; Zhuang, S.; Wu, Z.; Zhuang, Y.; Lin, Z.; Li, Z.; Li, D.; Xing, E.; et al. 2024. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36.

Zhou, R.; Fazel, M.; and Du, S. S. 2025. Extragradient preference optimization (egpo): Beyond last-iterate convergence for nash learning from human feedback. *arXiv preprint arXiv:2503.08942*.

Zhu, L.; Wang, X.; and Wang, X. 2025. JudgeLM: Fine-tuned Large Language Models are Scalable Judges. In *The Thirteenth International Conference on Learning Representations*.

Zou, D.; Wang, S.; Li, X.; Peng, H.; Wang, Y.; Liu, C.; Sheng, K.; and Zhang, B. 2024. MultiSPANS: A Multi-range Spatial-Temporal Transformer Network for Traffic Forecast via Structural Entropy Optimization. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, WSDM '24, 1032–1041. New York, NY, USA: Association for Computing Machinery. ISBN 9798400703713.