Localize-Then-Decide Guarantees for LLM Judgments

Xinyu Li[1], Yi Zhou[2], Guanqun Cao[3], Zeyu Fu[1], Tianjin Huang[1], Gaojie Jin[4,✉]
[1]University of Exeter [2]Cardiff University [3]University of the West of England
[4]University of Macau
Correspondence to: Gaojie Jin <gaojie.jin.kim@gmail.com>

Abstract

Large language models (LLMs) are increasingly used as evaluators to assess output quality and preference alignment, yet providing reliable guarantees of agreement with human judgments remains challenging. Recent work introduces confidence-thresholding methods that provide such guarantees for pairwise comparisons, relying on the assumption that higher estimated confidence implies lower disagreement risk with humans. However, this assumption can break down when the number of candidate responses increases, since distributing probability mass across many alternatives can distort confidence estimates. To address this issue, we propose a Localize-Then-Decide framework. First, conformal prediction localizes a small shortlist that contains the human-preferred response with high probability. Then, a calibrated confidence-based rule selectively chooses a single response from this shortlist or abstains. This design restores the monotonic relationship between confidence and disagreement risk and enables high-probability agreement guarantees. Experiments with multiple candidate sizes across several datasets and judge LLMs demonstrate that our framework consistently achieves higher guarantee success rates and substantially higher coverage than single-stage baselines.

1 Introduction

Large language models (LLMs) are increasingly deployed as evaluators for assessing output quality and preference alignment (Zheng et al., 2023; Dubois et al., 2023; Park et al., 2025; Chiang and Lee, 2023). Although this paradigm enables scalable and cost-effective alternatives to human annotation, a fundamental challenge persists (Xiong et al., 2023): how can decisions made by LLMs acting as judges be made reliably trustworthy for downstream use, particularly when the model expresses high confidence? To address this issue, Jung et al. (2024) derive exact disagreement-risk guarantees conditioned on a calibration set,

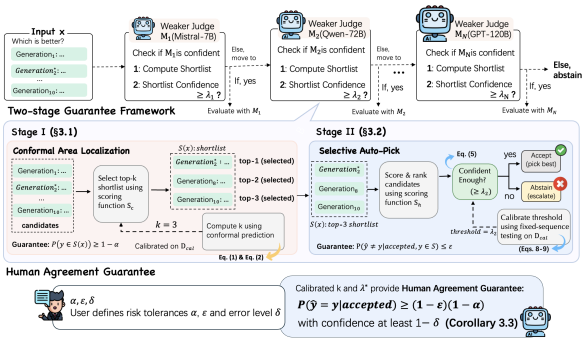

Figure 1: An overview of the Localize-Then-Decide (two-stage) guarantees framework.

but restrict attention to the pairwise generations ($m = 2$). In practical deployments, however, a judge LLM is typically required to select a single human-preferred response from multiple candidates ($m > 2$). Extending such reliability guarantees beyond the pairwise regime is therefore essential for real-world applicability.

A key assumption underlying the above confidence-thresholding guarantees (Jung et al., 2024) is monotonicity: instances assigned higher estimated confidence (probability) should exhibit lower disagreement risk with respect to human judgments. However, this assumption becomes problematic in the multi-candidate setting. When an LLM judge assigns preference probabilities over m candidates, the total probability mass must be distributed across all m options. As m increases, and the task typically becomes more challenging, the probability allocated to each individual candidate, including the true best, necessarily decreases as a direct consequence of normalization. This structural dilution effect can violate the monotonicity assumption and thereby complicates the direct extension of pairwise judgment guarantees to the multi-candidate regime. As shown in Figure 2(a), increasing m breaks the monotone confidence–agreement relationship; our experiments in Section 4 further corroborate this phenomenon.

Although large m may disrupt the monotonic-

Findings of the Association for Computational Linguistics: EMNLP 2026, pages 943–965
October 24-29, 2026 ©2026 Association for Computational Linguistics

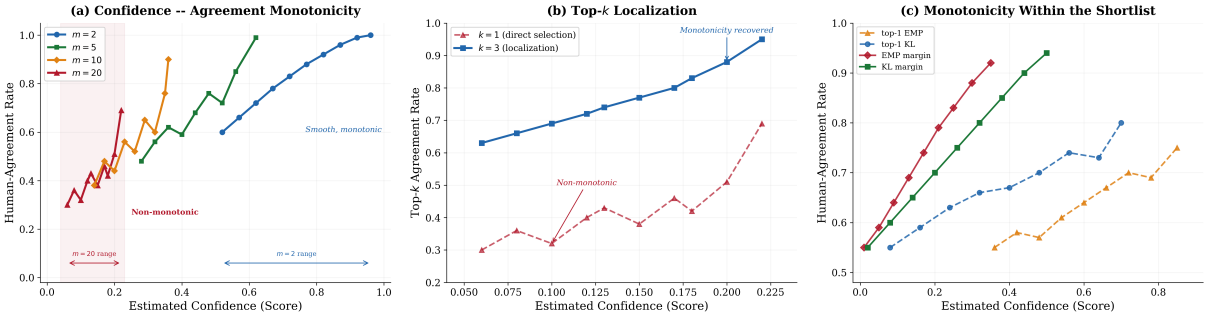

Figure 2: **Simulations on confidence–agreement monotonicity. (a)** As the number of candidates m increases, the monotone relationship between confidence scores and (above confidence) human-agreement rate progressively degrades. **(b)** Fixing m=20: although top-1 agreement loses monotonicity, top-3 agreement remains monotone with the same estimated confidence. **(c)** Within the localized (top-3) shortlist, the four estimated confidences (especially the margin ones) restore monotonicity. Details of the simulations are given in Appendix B.

ity of top-1 agreement with respect to confidence scores, we empirically observe that top-k agreement (e.g., top-3) and subsequent k-to-1 selection (e.g., selecting one from three) preserve a substantially more stable monotonic relationship with confidence measures. As shown in Figure 2(b), even when $m = 20$, the top-3 agreement rate remains monotone with respect to the estimated confidence. Furthermore, as shown in Figure 2(c), once the candidate set is restricted to the top-3 responses, the confidence–agreement relationship for selecting one out of three candidates becomes significantly more monotone (especially the margin-based confidence scores). These observations motivate a two-stage guarantee framework for aligning LLM judgments with human preferences. In the first stage, we employ conformal prediction to localize a small candidate set (top-k) that contains the human-preferred answer with high probability. In the second stage, we attempt to automatically select a single response from this localized set while providing a high-probability correctness guarantee; otherwise, the system abstains. We now present an informal summary of the resulting guarantee.

Theorem 1.1 (Informal) *In a multi-candidate setting, for user-specified risk tolerances α (Stage I: conformal area localization) and ε (Stage II: selective auto-picking), with confidence at least $1-\delta$, the proposed framework guarantees*

$$\mathbb{P}(\textit{LLM agrees with human} | \textit{Satisfied confidence}) \geq (1-\varepsilon)(1-\alpha).$$

Empirically, we demonstrate that, under the proposed framework, the scoring (confidence) functions exhibit a substantially improved monotonic relationship between confidence and disagreement risk, leading to higher success rates in achieving target human–LLM agreement levels across multiple datasets and judge LLMs. To summarize, the main contributions of this work are as follows:

- ⋆ **Practical Extension.** We extend existing human–LLM judgment guarantee frameworks from the pairwise setting ($m = 2$) to the more practical multi-generation setting ($m > 2$). We show that the presence of multiple candidates can cause estimated confidence scores to violate the monotonicity assumption that underlies confidence-thresholding guarantees.

- ⋆ **Two-Stage Guarantee.** To address this issue, we propose a two-stage guarantee framework for aligning LLM judgments with human preferences, motivated by the observation that such a decomposition restores the monotonic relationship between confidence and agreement. In the first stage, we employ conformal prediction to localize a small candidate set that contains the human-preferred response with high probability. In the second stage, we automatically select a single response from this localized set while providing a high-probability correctness guarantee; otherwise, the system abstains or escalates.

- ⋆ **Experiments.** We conduct extensive experiments across multiple datasets (e.g., TL;DR, Chatbot Arena, HH-RLHF, and AlpacaEval) and judge LLMs (e.g., Llama, Qwen, GPT-OSS, and DeepSeek) under various candidate-count settings ($m \in \{5, 10, 20\}$). The results demonstrate that our framework effectively restores the monotonic relationship between confidence and disagreement risk and significantly improves the reliability of human–LLM agreement guarantees.

2 Preliminary

2.1 Problem Setup

Let $f_{\mathrm{LM}} : \mathcal{X} \to \mathcal{Y}$ denote an LLM-based judge. Each input $x \in \mathcal{X}$ consists of a query together with m candidate responses $\{g_1, \ldots, g_m\}$, and the true label $y \in \mathcal{Y} = \{1, \ldots, m\}$ indicates the index of the response preferred by the human. We write $z = (x, y)$ for a labeled instance drawn from an unknown underlying distribution.

Given f_{LM}, we consider two real-valued scoring (confidence) functions: $\mathbb{S}_c(x, g_i)$, used in the first-stage conformal prediction procedure, and $\mathbb{S}_h(x, g_i)$, used in the second-stage hypothesis testing procedure. In both cases, larger scores indicate higher predicted preference for g_i, i.e., a greater likelihood of being the human-best response. Our objective is to output:

1. A small area (shortlist) $\mathcal{S}(x) \subseteq \{1, \ldots, m\}$ that contains the true label y with high probability;
2. Optionally, a single automatically selected index $\hat{y} \in \mathcal{S}(x)$ when such a choice can be certified as reliable. Otherwise, the system abstains or escalates the decision.

2.2 Assumptions

Our framework relies on the following standard and widely adopted assumptions.

Assumption 2.1 *Let $\mathcal{D} = \{z_i = (x_i, y_i)\}_{i=1}^n$ be a calibration set and z_{n+1} a future test point. Assume $(z_1, \ldots, z_n, z_{n+1})$ are exchangeable (i.e., i.i.d.).*

Assumption 2.2 *For every x and the corresponding $\{g_1, ..., g_m\}$, the human-best preference y is unique. Ties in ranking by scoring functions are broken deterministically.*

3 Two-Stage Certified Selection for Multi-Generations

3.1 Stage I: Conformal Area Localization

In this subsection, we describe a conformal prediction procedure for localizing a small area (shortlist) $\mathcal{S}(x)$ that contains the true label y with probability at least $1 - \alpha$. We first define a nonconformity score based on the rank of the human-best response under the scoring function $\mathbb{S}_c(x, \cdot)$:

$$R(z) \;=\; \big|\{i \in \{1, \ldots, m\} : \mathbb{S}_c(x, g_i) \geq \mathbb{S}_c(x, g_y)\}\big|, \quad (1)$$

which corresponds to the number of candidates whose score is no smaller than that of the human-best response. Given the calibration set, let k denote the conformal quantile defined as

$$k = \text{the } \lceil (n+1)(1-\alpha) \rceil \text{-th order statistic of } \{R(z_i)\}_{i=1}^n. \quad (2)$$

For a new test input x, we construct the area set

$$\mathcal{S}(x) = \{\text{indices of the top-}k \text{ candidates under } \mathbb{S}_c(x, \cdot)\}. \quad (3)$$

We now state the finite-sample coverage guarantee.

Theorem 3.1 (Finite-sample area guarantee) *Under Assumptions 2.1, 2.2, for any test input x, the area set $\mathcal{S}(x)$ constructed in (3) satisfies*

$$\mathbb{P}\big(y \in \mathcal{S}(x)\big) \;\geq\; 1 - \alpha. \quad (4)$$

Proof. See Appendix C. □

3.2 Stage II: Selective Auto-Pick

In this subsection, we seek to automatically select a single index from the localized area $\mathcal{S}(x)$ while providing a high-probability correctness guarantee; otherwise, the system abstains or escalates.

Let $\hat{y}$ denote the index predicted by the judge LLM f_{LM}. We define a margin-based confidence score within $\mathcal{S}(x)$ as

$$\Delta_{\mathcal{S}}(x) \;=\; \mathbb{S}_h(x, g_{\hat{y}}) \;-\; \max_{i \in \mathcal{S}(x),\, i \neq \hat{y}} \mathbb{S}_h(x, g_i). \quad (5)$$

Given a threshold λ, we define the conditional error rate of the auto-pick rule restricted to $\mathcal{S}(x)$ as

$$R_{\mathcal{S}}(\lambda) \;=\; \mathbb{P}(\hat{y} \neq y \mid \Delta_{\mathcal{S}}(x) \geq \lambda)\,. \quad (6)$$

Within $\mathcal{S}(x)$, observe that $\{\hat{y} \neq y\} \subseteq \{y \notin \mathcal{S}(x)\} \cup \{\hat{y} \neq y,\ y \in \mathcal{S}(x)\}$. Therefore, $\forall\ \lambda$,

$$\begin{aligned} R_{\mathcal{S}}(\lambda) \;\leq\; & \underbrace{\mathbb{P}(y \notin \mathcal{S}(x) \mid \Delta_{\mathcal{S}}(x) \geq \lambda)}_{\text{localization miss under acceptance}} \\ & + \underbrace{\mathbb{P}(\hat{y} \neq y \mid \Delta_{\mathcal{S}}(x) \geq \lambda,\ y \in \mathcal{S}(x))}_{\text{within-area selection error}}. \end{aligned} \quad (7)$$

In (7), the second term corresponds to the component that Stage II can directly calibrate, whereas the first term is partly controlled by the Stage I localization guarantee.

To calibrate the within-area selection error, we still use the calibration set $\mathcal{D} = \{z_j\}_{j=1}^n$. For a given threshold λ, define the set of accepted-and-covered indices

$$A^{\mathrm{in}}(\lambda) \;=\; \{j : \Delta_{\mathcal{S}}(x_j) \geq \lambda \text{ and } y_j \in \mathcal{S}(x_j)\},$$
$$N^{\mathrm{in}}(\lambda) = |A^{\mathrm{in}}(\lambda)|,$$

and the corresponding number of within-area errors

$$Y^{\mathrm{in}}(\lambda) = \sum_{j \in A^{\mathrm{in}}(\lambda)} \mathbf{1}\{\hat{y}_j \neq y_j\}.$$

Whenever $N^{\text{in}}(\lambda) \geq 1$, we define the empirical and population within-area error rates as

$$\widehat{R}^{\text{in}}_{\mathcal{S}}(\lambda) \;=\; \frac{Y^{\text{in}}(\lambda)}{N^{\text{in}}(\lambda)},$$
$$R^{\text{in}}_{\mathcal{S}}(\lambda) \;=\; \mathbb{P}(\hat{y} \neq y \mid \Delta_{\mathcal{S}}(x) \geq \lambda,\; y \in \mathcal{S}(x))\,.$$

Since $Y^{\text{in}}(\lambda) \mid N^{\text{in}}(\lambda) \sim \text{Bin}\big(N^{\text{in}}(\lambda), R^{\text{in}}_{\mathcal{S}}(\lambda)\big)$ under exchangeability, we define the exact $(1-\delta)$ upper confidence bound (UCB):

$$\widehat{R}^{\text{in},+}_{\mathcal{S}}(\lambda) \;:=\; \sup\Big\{r \in [0,1] : \mathbb{P}\big(\text{Bin}(N^{\text{in}}(\lambda), r) \leq Y^{\text{in}}(\lambda)\big) \;\geq\; \delta\Big\}. \tag{8}$$

Fixed-sequence threshold selection. Empirically, decreasing λ tends to near-monotonically increase risk (see Figures 2(c), 3(c)), because more ambiguous instances are accepted; i.e., $\widehat{R}^{\text{in},+}_{\mathcal{S}}(\lambda)$ tends to increase as λ decreases. This allows us to use fixed sequence testing (Bauer, 1991), wherein we test from the largest value of λ (e.g., 0.99) to a progressively smaller value, and stop at the last time $\widehat{R}^{\text{in},+}_{\mathcal{S}}$ is below the target risk ε, i.e.,

$$\lambda^* \;=\; \inf\Big\{\lambda : \widehat{R}^{\text{in},+}_{\mathcal{S}}(\lambda') \leq \varepsilon \;\text{ for all }\; \lambda' \geq \lambda\Big\}. \tag{9}$$

This is precisely the fixed-sequence testing rule adopted for selective evaluation to avoid Bonferroni conservatism in large hypothesis spaces. Then, we get the following guarantee.

Theorem 3.2 *Given Assumptions 2.1, 2.2, let λ^* be selected by the fixed-sequence rule (9) using the binomial upper confidence bound defined in (8), applied to the calibration subset*

$$\mathcal{D}_{\text{in}} := \{(x_i, y_i) \in \mathcal{D} \;:\; y_i \in \mathcal{S}(x_i)\}\,.$$

Then, with probability at least $1-\delta$ over the draw of the calibration sample, the following holds for a new example (x, y):

$$\mathbb{P}(\hat{y} \neq y \mid \Delta_{\mathcal{S}}(x) \geq \lambda^*, y \in \mathcal{S}(x)) \;\leq\; \varepsilon. \tag{10}$$

Proof. See Appendix C. □

Empirically, increasing λ approximately increases the top-k agreement rate (i.e., decreases $\mathbb{P}(y \notin \mathcal{S}(x) \mid \Delta_{\mathcal{S}}(x) \geq \lambda)$), as directly tested in Table 8. Under this empirically supported monotonicity assumption, we derive a composed bound for the selective error of the auto-pick decision. Stage I's marginal coverage guarantee is distribution-free under exchangeability, whereas the composed end-to-end guarantee below additionally relies on this cross-stage monotonicity condition.

Corollary 3.3 *Given Assumptions 2.1, 2.2, and the monotonicity assumption on $\mathbb{P}(y \notin \mathcal{S}(x) \mid \Delta_{\mathcal{S}}(x) \geq \lambda)$, let $\mathcal{S}(x)$ be produced by Stage I and λ^* calibrated by Stage II as in Theorem 3.2. Then, with probability at least $1-\delta$ over the calibration sample used in Stage II, we get*

$$\mathbb{P}\big(\hat{y} = y \mid \Delta_{\mathcal{S}}(x) \geq \lambda^*\big) \;\geq\; (1-\varepsilon)(1-\alpha).$$

Proof. See Appendix C. □

Remark. *Based on the monotonicity of top-k agreement (e.g., top-3) and the subsequent k-to-1 selection, we combine conformal prediction with fixed-sequence testing to obtain Corollary 3.3. The result yields an interpretable decomposition of the final selective accuracy guarantee into two factors: $(1-\alpha)$, the coverage probability of Stage I localization (i.e., the human-preferred response lies in the shortlist $\mathcal{S}(x)$), and $(1-\varepsilon)$, the conditional correctness of the Stage II decision rule on accepted instances. This multiplicative form highlights the role of the two-stage design: Stage I localizes reliable candidates, while Stage II controls decision risk through calibrated thresholding. Improving either shortlist coverage or conditional decision reliability directly strengthens the overall guarantee.*

4 Experiments

The previous sections establish the framework that decomposing m-to-1 selection into two stages to restore the monotonic confidence–agreement structure needed for valid guarantees. We now evaluate whether this theoretical benefit materializes on real-world benchmarks. Our experiments address three questions: **(Q1)** Does the monotonicity breakdown occur on real data, and does the two-stage decomposition recover it (Section 4.2)? **(Q2)** Does the two-stage framework yield stronger guarantees and higher coverage than single-stage baselines under multiple-m setting (Section 4.3)? **(Q3)** Can the framework extend to cascaded multi-model architectures to further improve coverage and guarantee success rate (Section 4.4)?

4.1 Experimental Setup

We first describe the datasets, models, and evaluation settings shared across all experiments.

Datasets. We evaluate our framework on four preference-evaluation benchmarks that cover diverse domains: **(1) TL;DR** (Stiennon et al., 2020), a summarization dataset where the judge selects the best summary from m candidates; **(2) Chatbot Arena** (Zheng et al., 2023), comprising real-world user interactions with chatbot systems; **(3) HH-RLHF** (Bai et al., 2022), covering helpfulness and harmlessness preferences; **(4) AlpacaEval** (Dubois et al., 2023), an instruction-following evaluation benchmark. For each dataset we construct multi-candidate instances with $m \in \{5, 10, 20\}$ candidates per query, yielding approximately 3,000 instances per $(dataset, m)$ pair. All datasets include human-annotated ground-truth preferences. The detailed procedure for constructing multi-candidate instances is provided in Appendix D.

Judge Models. We employ a range of open-weight LLMs spanning different parameter scales, enabling both single-model evaluation and multi-model cascading: **Mistral-7B-Instruct-v0.2** (7B), **Llama-3-8B-Instruct** (8B), **Qwen2.5-7B-Instruct** (7B), **DeepSeek-V2-Lite-Chat** (16B), **Qwen2.5-32B-Instruct** (32B), **DeepSeek-67B-Chat** (67B), **Llama-3-70B-Instruct** (70B), **Qwen2.5-72B-Instruct** (72B), and **GPT-OSS-120B** (120B). This covers five model families (Mistral, Llama, Qwen, DeepSeek, GPT) with a parameter-size gradient from 7B to 120B, provides sufficient diversity for constructing multiple cascaded configurations. All judge models are evaluated on the same setting of instances per dataset, ensuring data alignment across models so that any cascade configuration can be composed without missing instances.

Confidence Estimation. Following Jung et al. (2024), we use Simulated Annotators with K=5 few-shot examples per N=5 simulated annotators. For each instance, we compute and store the full output probability distributions, allowing all downstream scoring functions to be evaluated directly.

Scoring (Confidence) Functions. For Stage I conformal localization, we adopt the Ensemble Mean Probability (EMP) as the scoring function $\mathbb{S}_c$. For Stage II selective auto-picking, we use the KL-divergence as the default scoring function $\mathbb{S}_h$, and compute the margin according to (5). We also evaluate several alternative scoring functions in Stage II, including non-margin variants that directly use the raw score without computing the margin in (5). The full definitions of all scoring functions are provided in Appendix E.

Evaluation Protocol. For each (dataset, model) combination, we repeat the following procedure 1,000 times: randomly split the data into calibration and test sets with a (50%/50%) ratio, apply our framework (or the single-stage baseline of Jung et al. (2024)) using calibration set, and evaluate performance on test set. We report two metrics:

- **Coverage Rate (Cov.)**: the proportion of test instances that are accepted by the system, averaged over the 1,000 random splits. Higher values indicate that the method can safely make decisions on a larger fraction of inputs.
- **Guarantee Success Rate (GSR)**: the proportion of the 1,000 splits in which the empirical human–LLM agreement on accepted test instances exceeds the target guarantee level (for our two-stage framework, this is $(1-\varepsilon)(1-\alpha)$). Ideally, this proportion should be at least the confidence level of the guarantee, i.e., $1-\delta$.

4.2 Validating Monotonicity across Datasets and Judge LLMs

The simulations in Figure 2 suggest that single-stage monotonicity degrades with m but can be recovered by two-stage decomposition. We now quantify this phenomenon on real datasets using the ranking loss across multiple models and datasets.

Monotonicity metric. Given T evenly spaced confidence thresholds $\lambda_1 > \cdots > \lambda_T$ and the corresponding agreement rates $a_1, \ldots, a_T$, monotonicity requires $a_1 \geq \cdots \geq a_T$. We measure violations by the ranking loss: $\mathcal{L}_{\text{rank}} = \sum_{i<j} \mathbf{1}[a_i < a_j] / \binom{T}{2}$, where $\mathcal{L}_{\text{rank}}$=0 indicates perfect monotonicity and $\mathcal{L}_{\text{rank}}\approx$0.5 indicates random ordering. We evaluate three settings corresponding to the panels of Figure 2: Direct m-to-1, Stage I (m-to-k), and Stage II (k-to-1). Details of the setting are given in Appendix G.

Table 1 reports ranking loss across four datasets, three judge models, and three candidate-set sizes. **(1)** Single-stage ranking loss increases substantially with m (e.g., $0.14 \rightarrow 0.34$ for Qwen2.5-72B on TL;DR), confirming the monotonicity breakdown shown in Figure 2(a). **(2)** Both Stage I and Stage II maintain low ranking loss (≤ 0.08) across

Dataset	Judge	Direct m-to-1			Stage I (m-to-k)			Stage II (k-to-1)		
		m=5	m=10	m=20	m=5	m=10	m=20	m=5	m=10	m=20
TL;DR	Llama-3-8B	0.21	0.32	0.43	0.03	0.05	0.07	0.02	0.04	0.05
	Qwen2.5-72B	0.14	0.24	0.34	0.01	0.02	0.04	0.00	0.01	0.02
	GPT-OSS-120B	0.11	0.20	0.30	0.01	0.02	0.03	0.00	0.01	0.02
Chatbot Arena	Llama-3-8B	0.24	0.35	0.46	0.04	0.06	0.08	0.03	0.05	0.07
	Qwen2.5-72B	0.15	0.26	0.37	0.02	0.03	0.05	0.01	0.02	0.03
	GPT-OSS-120B	0.12	0.22	0.33	0.01	0.02	0.04	0.01	0.02	0.03
HH-RLHF	Llama-3-8B	0.19	0.30	0.40	0.03	0.05	0.07	0.02	0.03	0.05
	Qwen2.5-72B	0.14	0.21	0.31	0.01	0.02	0.03	0.00	0.01	0.02
	GPT-OSS-120B	0.10	0.18	0.27	0.01	0.01	0.03	0.00	0.01	0.01
AlpacaEval	Llama-3-8B	0.22	0.34	0.45	0.04	0.06	0.08	0.03	0.05	0.06
	Qwen2.5-72B	0.14	0.23	0.35	0.01	0.02	0.04	0.00	0.02	0.03
	GPT-OSS-120B	0.11	0.19	0.31	0.01	0.02	0.03	0.00	0.01	0.02

Table 1: **Ranking loss ($\mathcal{L}_{\text{rank}}$, ↓) measuring confidence–agreement monotonicity on real data** (T=20 thresholds, α=0.10). Direct m-to-1: single-stage selection using top-1 EMP confidence. Stage I (m-to-k): top-k localization agreement using top-1 EMP confidence. Stage II (k-to-1): selection within the shortlist using KL margin.

all settings, validating that the two-stage decomposition restores the monotonic structure required for reliable threshold calibration. The confidence–agreement curves in Figure 3 visually confirm this contrast: the single-stage curves degrade with m while both stages of our decomposition remain monotonic. Furthermore, we provide confidence-score distributions (see Appendix G) for additional supporting evidence.

4.3 Two-Stage vs. Single-Stage

Having confirmed that the two-stage decomposition restores monotonicity, we now test whether this translates into practical guarantee improvements. We use three scoring (confidence) functions: EMP margin, KL margin, and Vote agreement (defined in Appendix E), with both a **Single-Stage** and a **Two-Stage** framework, along with No Selection and Single-Stage (Jung et al., 2024) as references.

Two-stage achieves valid guarantees with higher coverage. Table 2 reports coverage and GSR at m=10 (target agreement = 0.81, δ=0.10) across four datasets and three representative judges from different model families (8B–120B), averaged over 1,000 random splits. Without any selection, even the strongest judge (GPT-OSS-120B) achieves only 54–60% agreement, well below the 81% target, confirming that selective evaluation is necessary. Across all scoring functions and datasets, every two-stage variant exceeds the $1-\delta = 90\%$ GSR target (e.g., 95–97% for KL margin on GPT-OSS-120B), while its single-stage counterpart using the same scoring function remains far below (48–79%). The improvement extends to coverage as well: Two-Stage (KL margin) with GPT-OSS-120B covers 64–72% of instances versus 50–58% for Single-Stage (KL margin), and even for the smallest judge (Llama-3-8B) two-stage raises coverage by 8–10%. Since matched Single-/Two-Stage pairs differ only in Stage I, the gain is architecture-driven rather than scoring-specific. Targeted comparisons with a variance-normalized adaptation of Jung et al. (2024) and component-wise KL-margin ablations are provided in Appendix H.

The performance of two-stage framework is robust across target levels, candidate-set sizes, and random splits. Figure 4(a) sweeps the target agreement continuously from 70% to 95% at m=10, showing whether the guarantee holds across the full target range rather than at one point. The two-stage empirical agreement (blue) tracks the target throughout, whereas the single-stage curve (red) falls below for most targets (guarantee fails). The shaded bands (min/max over 1,000 splits) further confirm that two-stage calibration is substantially more stable (e.g., $\pm$2.1% vs. $\pm$4.7% spread at target 85% for Qwen2.5-72B). Figure 4(b) directly verifies our core motivation by varying m from 5 to 20: single-stage GSR degrades sharply from 82% to 50%, showing that the single-stage failure is caused by increasing m; in contrast, two-stage GSR remains above the $1-\delta$=90% threshold throughout, validating the robustness of our two-stage framework against the m increasing. More results for all judges are provided in Appendix I.

4.4 Cascaded Architecture

The results above establish that our two-stage framework provides valid guarantees for any individual judge. In practice, however, deploying a

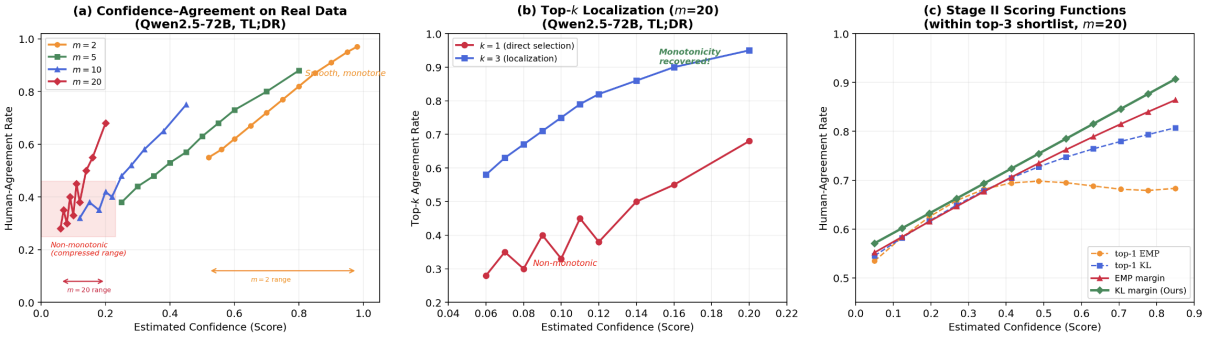

Figure 3: **Confidence–agreement curves on TL;DR and Qwen2.5-72B. (a)** Single-stage m-to-1. **(b)** Stage I top-k localization. **(c)** Stage II scoring within the top-3 shortlist (m=20).

Judge	Method	TL;DR			Chatbot Arena			HH-RLHF			AlpacaEval		
		Cov.	GSR	Agr.	Cov.	GSR	Agr.	Cov.	GSR	Agr.	Cov.	GSR	Agr.
Llama-3 8B	No Selection	100%	0%	$36\%_{\pm 1.2}$	100%	0%	$31\%_{\pm 1.4}$	100%	0%	$40\%_{\pm 1.1}$	100%	0%	$33\%_{\pm 1.3}$
	Single-Stage (top-1 EMP)	$26\%_{\pm 3.2}$	53.8%	$77\%_{\pm 2.3}$	$21\%_{\pm 3.6}$	47.6%	$74\%_{\pm 2.5}$	$30\%_{\pm 2.9}$	58.4%	$79\%_{\pm 2.0}$	$25\%_{\pm 3.4}$	52.1%	$76\%_{\pm 2.2}$
	Single-Stage (EMP margin)	$31\%_{\pm 3.0}$	58.6%	$78\%_{\pm 2.1}$	$26\%_{\pm 3.4}$	53.2%	$76\%_{\pm 2.3}$	$35\%_{\pm 2.7}$	63.5%	$80\%_{\pm 1.9}$	$28\%_{\pm 3.2}$	55.8%	$78\%_{\pm 2.2}$
	Single-Stage (KL margin)	$35\%_{\pm 2.8}$	63.1%	$80\%_{\pm 2.0}$	$30\%_{\pm 3.1}$	57.8%	$78\%_{\pm 2.1}$	$39\%_{\pm 2.5}$	68.3%	$82\%_{\pm 1.8}$	$33\%_{\pm 3.0}$	61.7%	$79\%_{\pm 2.1}$
	Single-Stage (Vote)	$33\%_{\pm 2.9}$	60.7%	$79\%_{\pm 2.0}$	$28\%_{\pm 3.2}$	55.4%	$77\%_{\pm 2.2}$	$37\%_{\pm 2.7}$	65.2%	$81\%_{\pm 1.8}$	$31\%_{\pm 3.0}$	59.3%	$79\%_{\pm 2.0}$
	Two-Stage (EMP margin)	$39\%_{\pm 2.6}$	**91.2%**	$84\%_{\pm 1.6}$	$34\%_{\pm 2.9}$	**90.4%**	$82\%_{\pm 1.8}$	$43\%_{\pm 2.4}$	**92.6%**	$85\%_{\pm 1.4}$	$37\%_{\pm 2.8}$	**90.8%**	$84\%_{\pm 1.7}$
	Two-Stage (KL margin)	$45\%_{\pm 2.3}$	**95.1%**	$86\%_{\pm 1.4}$	$40\%_{\pm 2.6}$	**93.7%**	$85\%_{\pm 1.6}$	$49\%_{\pm 2.2}$	**96.4%**	$87\%_{\pm 1.2}$	$43\%_{\pm 2.5}$	**94.3%**	$86\%_{\pm 1.5}$
	Two-Stage (Vote)	$42\%_{\pm 2.5}$	**93.4%**	$85\%_{\pm 1.5}$	$37\%_{\pm 2.7}$	**91.8%**	$84\%_{\pm 1.7}$	$46\%_{\pm 2.3}$	**94.1%**	$86\%_{\pm 1.3}$	$40\%_{\pm 2.6}$	**92.5%**	$85\%_{\pm 1.6}$
Qwen2.5 72B	No Selection	100%	0%	$52\%_{\pm 1.1}$	100%	0%	$48\%_{\pm 1.3}$	100%	0%	$54\%_{\pm 1.0}$	100%	0%	$51\%_{\pm 1.2}$
	Single-Stage (top-1 EMP)	$41\%_{\pm 2.7}$	62.4%	$80\%_{\pm 1.9}$	$35\%_{\pm 3.0}$	56.8%	$78\%_{\pm 2.1}$	$43\%_{\pm 2.5}$	65.7%	$82\%_{\pm 1.7}$	$40\%_{\pm 2.8}$	61.3%	$80\%_{\pm 2.0}$
	Single-Stage (EMP margin)	$46\%_{\pm 2.5}$	67.3%	$82\%_{\pm 1.8}$	$40\%_{\pm 2.8}$	62.1%	$80\%_{\pm 2.0}$	$48\%_{\pm 2.4}$	70.8%	$83\%_{\pm 1.6}$	$45\%_{\pm 2.6}$	66.5%	$81\%_{\pm 1.9}$
	Single-Stage (KL margin)	$50\%_{\pm 2.4}$	72.6%	$83\%_{\pm 1.7}$	$44\%_{\pm 2.6}$	67.4%	$81\%_{\pm 1.9}$	$52\%_{\pm 2.3}$	75.3%	$84\%_{\pm 1.5}$	$49\%_{\pm 2.5}$	71.2%	$83\%_{\pm 1.7}$
	Single-Stage (Vote)	$48\%_{\pm 2.5}$	69.8%	$82\%_{\pm 1.8}$	$42\%_{\pm 2.7}$	64.6%	$80\%_{\pm 1.9}$	$50\%_{\pm 2.3}$	73.1%	$84\%_{\pm 1.6}$	$47\%_{\pm 2.5}$	68.7%	$82\%_{\pm 1.8}$
	Two-Stage (EMP margin)	$57\%_{\pm 2.1}$	**92.3%**	$86\%_{\pm 1.3}$	$51\%_{\pm 2.4}$	**90.7%**	$85\%_{\pm 1.5}$	$59\%_{\pm 2.0}$	**93.1%**	$87\%_{\pm 1.1}$	$56\%_{\pm 2.2}$	**91.6%**	$86\%_{\pm 1.4}$
	Two-Stage (KL margin)	$64\%_{\pm 1.9}$	**95.8%**	$88\%_{\pm 1.1}$	$58\%_{\pm 2.2}$	**94.6%**	$87\%_{\pm 1.3}$	$66\%_{\pm 1.8}$	**96.2%**	$89\%_{\pm 1.0}$	$63\%_{\pm 2.0}$	**95.1%**	$88\%_{\pm 1.2}$
	Two-Stage (Vote)	$61\%_{\pm 2.0}$	**94.2%**	$87\%_{\pm 1.2}$	$55\%_{\pm 2.3}$	**92.9%**	$86\%_{\pm 1.4}$	$63\%_{\pm 1.9}$	**94.7%**	$88\%_{\pm 1.0}$	$60\%_{\pm 2.1}$	**93.6%**	$87\%_{\pm 1.3}$
GPT-OSS 120B	No Selection	100%	0%	$58\%_{\pm 1.0}$	100%	0%	$54\%_{\pm 1.2}$	100%	0%	$60\%_{\pm 0.9}$	100%	0%	$57\%_{\pm 1.1}$
	Single-Stage (top-1 EMP)	$47\%_{\pm 2.5}$	66.3%	$82\%_{\pm 1.8}$	$41\%_{\pm 2.8}$	61.4%	$80\%_{\pm 2.0}$	$49\%_{\pm 2.3}$	69.7%	$83\%_{\pm 1.6}$	$46\%_{\pm 2.6}$	65.2%	$81\%_{\pm 1.9}$
	Single-Stage (EMP margin)	$52\%_{\pm 2.3}$	71.8%	$83\%_{\pm 1.7}$	$46\%_{\pm 2.6}$	66.3%	$81\%_{\pm 1.9}$	$54\%_{\pm 2.1}$	74.6%	$85\%_{\pm 1.4}$	$51\%_{\pm 2.4}$	70.4%	$83\%_{\pm 1.7}$
	Single-Stage (KL margin)	$56\%_{\pm 2.1}$	76.4%	$85\%_{\pm 1.5}$	$50\%_{\pm 2.4}$	71.7%	$83\%_{\pm 1.7}$	$58\%_{\pm 2.0}$	79.3%	$86\%_{\pm 1.3}$	$55\%_{\pm 2.2}$	74.8%	$84\%_{\pm 1.6}$
	Single-Stage (Vote)	$54\%_{\pm 2.2}$	73.6%	$84\%_{\pm 1.6}$	$48\%_{\pm 2.5}$	68.9%	$82\%_{\pm 1.8}$	$56\%_{\pm 2.0}$	76.8%	$85\%_{\pm 1.4}$	$53\%_{\pm 2.3}$	72.3%	$84\%_{\pm 1.6}$
	Two-Stage (EMP margin)	$63\%_{\pm 1.9}$	**93.4%**	$87\%_{\pm 1.2}$	$57\%_{\pm 2.2}$	**91.8%**	$86\%_{\pm 1.4}$	$65\%_{\pm 1.8}$	**94.3%**	$88\%_{\pm 1.0}$	$62\%_{\pm 2.0}$	**92.7%**	$87\%_{\pm 1.3}$
	Two-Stage (KL margin)	$70\%_{\pm 1.7}$	**96.8%**	$89\%_{\pm 1.0}$	$64\%_{\pm 2.0}$	**95.4%**	$88\%_{\pm 1.2}$	$72\%_{\pm 1.6}$	**97.1%**	$90\%_{\pm 0.9}$	$69\%_{\pm 1.8}$	**96.3%**	$89\%_{\pm 1.1}$
	Two-Stage (Vote)	$67\%_{\pm 1.8}$	**95.2%**	$88\%_{\pm 1.1}$	$61\%_{\pm 2.1}$	**93.6%**	$87\%_{\pm 1.3}$	$69\%_{\pm 1.7}$	**95.7%**	$89\%_{\pm 0.9}$	$66\%_{\pm 1.9}$	**94.8%**	$88\%_{\pm 1.2}$

Table 2: **Two-stage vs. single-stage across judge models and datasets** (m=10, target agreement = 0.81, δ=0.10). Two-stage: ε=α=0.10; single-stage: ε'=0.19. For Two-Stage framework, Stage I always uses EMP scores. Agr. denotes average empirical human-agreement on accepted instances. Cov. and Agr. are reported as mean $\pm$ std over 1,000 random splits. We provide more results in Appendix I.

single large model for all instances is expensive. A natural strategy is to cascade judges of increasing capacity: a weaker model evaluates first; only when it abstains does the instance escalate to a stronger one. Jung et al. (2024) demonstrate this for m=2, but single-stage guarantees break down for $m > 2$ (Section 4.3). Because our two-stage framework restores valid per-tier guarantees for $m > 2$, it naturally enables cascaded architectures with provable reliability. We evaluate five cascade configurations (Table 3), including same-family, cross-family, and two-tier setups, along with two single-model references, all under the same protocol as Section 4.3. Each tier operates as a complete two-stage selective evaluator; an instance is accepted only when the tier's Stage II confidence exceeds its calibrated threshold, and escalated otherwise.

Two-stage cascades maintain guarantees; single-stage cascades do not. Table 3 compares both variants side by side. Without two-stage decomposition, every cascade fails to reach the $1-\delta = 90\%$ GSR threshold (64–76%), consistent with the per-model failures in Table 2; simply adding more models does not fix the monotonicity breakdown at each tier. In contrast, every two-stage cascade exceeds 90% GSR (91–95%), confirming that valid per-tier guarantees compose correctly. Cascading also substantially improves coverage: the Qwen same-family cascade reaches 78% versus 64% for Qwen2.5-72B alone (+14%), and across all three-tier configurations two-stage cascades cover 77–81% of instances compared with 64–70% for the strongest single model. The composition columns show that the strongest model is invoked for fewer than 30% of instances in every three-tier cascade, demonstrating significant computational savings.

Guarantee curve and evaluator composition. Figure 4(c)–(d) illustrates a representative cascade

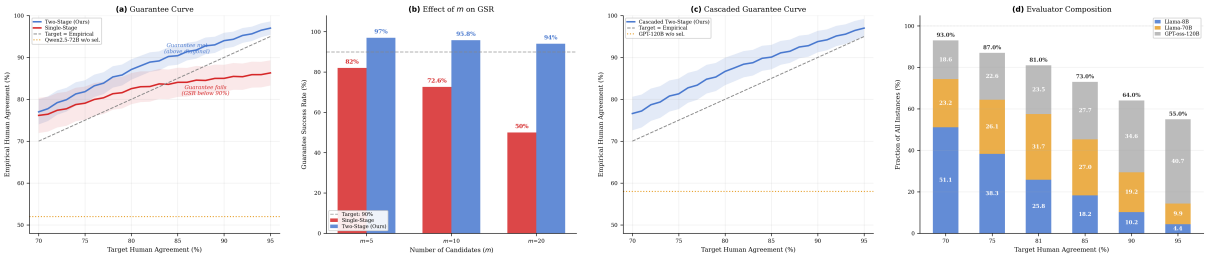

Figure 4: **Guarantee and cascade results** (TL;DR dataset). **(a)** Single-model guarantee curve (Qwen2.5-72B, m=10): Two-Stage (blue) consistently meets the target (above diagonal); Single-Stage (red) falls below. Shaded bands show min/max over 1,000 splits. **(b)** Effect of m on GSR (Qwen2.5-72B): Single-Stage drops from 82% to 50% as m grows; Two-Stage remains $\geq 91\%$. **(c)** Cascaded guarantee curve (Llama-8B $\rightarrow$ 70B $\rightarrow$ GPT-OSS-120B, m=10): empirical agreement tracks the target throughout. **(d)** Evaluator composition (same cascade as (c)) as fraction of all test instances: at lower targets the weakest tier handles most instances; as the target tightens, more instances escalate to stronger tiers.

Cascade Configuration	Single-Stage		**Two-Stage (Ours)**				
	Cov.	GSR	Cov.	GSR	Comp. (%)		
					T_1	T_2	T_3
Qwen2.5-72B only	50% $\pm$2.4	72.6%	64% $\pm$1.9	95.8%	–	–	100.0
GPT-oss-120B only	56% $\pm$2.1	76.4%	70% $\pm$1.7	96.8%	–	–	100.0
Qwen-7B $\rightarrow$ 32B $\rightarrow$ 72B	60% $\pm$2.8	65.7%	78% $\pm$1.9	**93.1%**	37.8	35.6	26.5
Llama-8B $\rightarrow$ 70B $\rightarrow$ GPT-120B	63% $\pm$2.5	68.3%	81% $\pm$1.8	**94.2%**	31.9	39.1	29.0
Mistral-7B $\rightarrow$ DS-67B $\rightarrow$ GPT-120B	58% $\pm$3.1	64.8%	77% $\pm$2.2	**91.6%**	35.3	38.2	26.5
Llama-8B $\rightarrow$ Qwen-32B $\rightarrow$ 72B	61% $\pm$2.7	66.9%	79% $\pm$2.0	**92.8%**	34.9	38.1	27.0
DeepSeek-16B $\rightarrow$ Qwen-72B	57% $\pm$3.2	63.5%	74% $\pm$2.3	**91.4%**	45.7	54.3	–

Table 3: **Cascaded evaluation on TL;DR** (m=10, target agreement = 0.81, δ=0.10). Instances abstained by one tier escalate to the next. T_i denotes the i-th model in the cascade (weakest first); "Comp." shows each tier's share of the covered instances (summing to 100%). Coverage and GSR are reported as mean $\pm$ std over 1,000 random splits.

(Llama-8B $\rightarrow$ 70B $\rightarrow$ GPT-120B). In (c), empirical agreement tracks the target over the 70–95% range, showing that the cascaded guarantee remains stable across target levels. Panel (d) shows that the cascade covers 93.0% of instances at the 70% target, with the weakest tier handling over half; as the target tightens, total coverage decreases and more instances are routed to stronger tiers. Additional cascade results and related work are provided in Appendices A and I.

5 Conclusion

In this work, we studied the problem of providing reliable human–LLM agreement guarantees for LLM-based evaluators in realistic multi-candidate judgment settings, where a judge must select a single preferred response from several alternatives. While prior confidence-thresholding approaches provide guarantees in the pairwise case, their key monotonicity assumption between confidence and disagreement risk breaks down as the number of candidates increases, making existing guarantees unreliable in practical evaluation pipelines. To address this limitation, we introduced a Localize-Then-Decide framework, which decomposes the multi-candidate decision into two calibrated stages. A conformal prediction step first localizes a small shortlist that contains the human-preferred response with high probability, and a selective decision rule then chooses a single response from this shortlist or abstains when reliability cannot be ensured. This decomposition restores the confidence–agreement monotonicity required for valid threshold calibration and yields finite-sample guarantees on selective human–LLM agreement. Empirical results across multiple datasets, judge models, and candidate sizes demonstrate that the proposed framework consistently achieves valid guarantees while improving coverage compared with single-stage baselines. In addition, the framework naturally enables cascaded evaluation architectures, allowing weaker models to handle easy cases while escalating difficult instances to stronger models without violating reliability guarantees.

Overall, our theoretical and empirical results suggest that structuring LLM judgment as a localized and selectively certified decision process is key to achieving trustworthy automated evaluation. By extending provable guarantees from pairwise comparisons to realistic multi-generation scenarios, this work takes an important step toward reliable and scalable LLM-as-a-judge systems.

Limitations

As with all conformal-prediction-based methods, our theoretical guarantees require the calibration and test instances to be exchangeable (Assumption 2.1). When the evaluation domain at deployment time diverges from the calibration corpus (e.g., due to topic shift or prompt reformulation), the finite-sample bounds may become less tight (Gibbs and Candes, 2021). While fully relaxing this assumption remains an open problem shared by the broader conformal prediction literature, our consistent results across four benchmarks spanning diverse domains (summarization, open-ended dialogue, safety, and instruction-following) suggest that the framework is applicable to a wide variety of evaluation scenarios when calibration data is drawn from the target domain.

Our confidence estimation builds on Simulated Annotators (Jung et al., 2024), requiring N forward passes per instance (5 in our default configuration), each using a prompt with $K{=}5$ few-shot examples; this is more expensive than obtaining a single predictive probability. However, the two-stage framework itself is agnostic to the choice of confidence estimator; Simulated Annotators can be replaced by lighter-weight alternatives (e.g., a single-pass predictive probability or verbalized confidence) whenever the resulting scores preserve sufficient discriminative power for threshold calibration.

Finally, our evaluation focuses on the preference-judgment setting where the judge selects a single preferred response from multiple candidates. While this covers a wide range of practical LLM evaluation scenarios (e.g., best-of-N decoding, multi-system comparison), other evaluation paradigms such as Likert-scale scoring or factuality verification involve different output structures and may require task-specific adaptations of the scoring and calibration procedures. Nonetheless, the core Localize-Then-Decide principle is general and could serve as a foundation for extending formal guarantees to these broader settings.

Ethics Statement

This work aims to improve the reliability of LLM-based evaluation by providing formal statistical guarantees on human–LLM agreement. All experiments use publicly available datasets with existing human-preference annotations; no new human data was collected for this study. We note that the guarantees provided by our framework are statistical in nature and hold under the stated assumptions. They should not be interpreted as absolute correctness certificates, and human oversight remains important in high-stakes evaluation scenarios.

Acknowledgments

This work was supported by the NVIDIA Academic Grant Program (Exploiting Overthinking Attacks on GenAI), the Royal Society Grant (Ensuring Trustworthy AI: Robustness Certification for Large Language Models) [Reference RGS\R2\252444], and the AIRR Gateway project (Exploiting Robustness of Reasoning Efficiency in Agentic AI).

References

Anastasios N Angelopoulos and Stephen Bates. 2023. Conformal prediction: A gentle introduction. *Foundations and Trends in Machine Learning*, 16(4):494–591.

Anastasios N Angelopoulos, Stephen Bates, Emmanuel J Candès, Michael I Jordan, and Lihua Lei. 2025. Learn then test: Calibrating predictive algorithms to achieve risk control. *The Annals of Applied Statistics*, 19(2):1641–1662.

Anastasios N Angelopoulos, Stephen Bates, Adam Fisch, Lihua Lei, and Tal Schuster. 2022. Conformal risk control. *arXiv preprint arXiv:2208.02814*.

Sher Badshah, Ali Emami, and Hassan Sajjad. 2026. Scope: Selective conformal optimized pairwise llm judging. *arXiv preprint arXiv:2602.13110*.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, and 1 others. 2022. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*.

Stephen Bates, Anastasios Angelopoulos, Lihua Lei, Jitendra Malik, and Michael Jordan. 2021. Distribution-free, risk-controlling prediction sets. *Journal of the ACM (JACM)*, 68(6):1–34.

Peter Bauer. 1991. Multiple testing in clinical trials. *Statistics in medicine*, 10(6):871–890.

Guiming Hardy Chen, Shunian Chen, Ziche Liu, Feng Jiang, and Benyou Wang. 2024. Humans or llms as the judge? a study on judgement bias. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 8301–8327.

Cheng-Han Chiang and Hung-yi Lee. 2023. Can large language models be an alternative to human evaluations? In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15607–15631.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, and 1 others. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Gianluca Detommaso, Martin Bertran, Riccardo Fogliato, and Aaron Roth. 2024. Multicalibration for confidence scoring in llms. *arXiv preprint arXiv:2404.04689*.

Yi Dong, Ronghui Mu, Gaojie Jin, Yi Qi, Jinwei Hu, Xingyu Zhao, Jie Meng, Wenjie Ruan, and Xiaowei Huang. 2024. Position: Building guardrails for large language models requires systematic design. In *International Conference on Machine Learning*.

Yann Dubois, Chen Xuechen Li, Rohan Taori, Tianyi Zhang, Ishaan Gulrajani, Jimmy Ba, Carlos Guestrin, Percy S Liang, and Tatsunori B Hashimoto. 2023. Alpacafarm: A simulation framework for methods that learn from human feedback. *Advances in Neural Information Processing Systems*, 36:30039–30069.

Ran El-Yaniv and 1 others. 2010. On the foundations of noise-free selective classification. *Journal of Machine Learning Research*, 11(5).

Yonatan Geifman and Ran El-Yaniv. 2017. Selective classification for deep neural networks. *Advances in neural information processing systems*, 30.

Jiahui Geng, Fengyu Cai, Yuxia Wang, Heinz Koeppl, Preslav Nakov, and Iryna Gurevych. 2024. A survey of confidence estimation and calibration in large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6577–6595.

Isaac Gibbs and Emmanuel Candes. 2021. Adaptive conformal inference under distribution shift. *Advances in Neural Information Processing Systems*, 34:1660–1672.

Yu Gui, Ying Jin, and Zhimei Ren. 2024. Conformal alignment: Knowing when to trust foundation models with guarantees. *Advances in Neural Information Processing Systems*, 37:73884–73919.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. 2017. On calibration of modern neural networks. In *International conference on machine learning*, pages 1321–1330. PMLR.

Gaojie Jin, Yong Tao, Lijia Yu, and Tianjin Huang. 2026. Margin-adaptive confidence ranking for reliable LLM judgement. In *International Conference on Machine Learning*.

Gaojie Jin, Sihao Wu, Jiaxu Liu, Tianjin Huang, and Ronghui Mu. 2025. Enhancing robust fairness via confusional spectral regularization. In *International Conference on Learning Representations*, volume 2025.

Gaojie Jin, Xinping Yi, Wei Huang, Sven Schewe, and Xiaowei Huang. 2022. Enhancing adversarial training with second-order statistics of weights. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*.

Gaojie Jin, Xinping Yi, Liang Zhang, Lijun Zhang, Sven Schewe, and Xiaowei Huang. 2020. How does weight correlation affect generalisation ability of deep neural networks? *Advances in Neural Information Processing Systems*.

Jaehun Jung, Faeze Brahman, and Yejin Choi. 2024. Trust or escalate: Llm judges with provable guarantees for human agreement. *arXiv preprint arXiv:2407.18370*.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, and 1 others. 2022. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*.

Reza Khanmohammadi, Erfan Miahi, Mehrsa Mardikoraem, Simerjot Kaur, Ivan Brugere, Charese Smiley, Kundan S Thind, and Mohammad M Ghassemi. 2025. Calibrating llm confidence by probing perturbed representation stability. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 10459–10525.

Seungone Kim, Jay Shin, Joel Jang, Shayne Longpre, Hwaran Lee, Sangdoo Yun, Ryan Shin, Sungdong Kim, James Thorne, Minjoon Seo, and 1 others. 2024a. Prometheus: Inducing fine-grained evaluation capability in language models. In *International Conference on Learning Representations*, volume 2024, pages 29927–29962.

Seungone Kim, Juyoung Suk, Shayne Longpre, Bill Yuchen Lin, Jamin Shin, Sean Welleck, Graham Neubig, Moontae Lee, Kyungjae Lee, and Minjoon Seo. 2024b. Prometheus 2: An open source language model specialized in evaluating other language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 4334–4353.

Bhawesh Kumar, Charlie Lu, Gauri Gupta, Anil Palepu, David Bellamy, Ramesh Raskar, and Andrew Beam. 2023. Conformal prediction with large language models for multi-choice question answering. *arXiv preprint arXiv:2305.18404*.

Nathan Lambert, Valentina Pyatkin, Jacob Morrison, Lester James Validad Miranda, Bill Yuchen Lin, Khyathi Chandu, Nouha Dziri, Sachin Kumar, Tom Zick, Yejin Choi, and 1 others. 2025. Rewardbench: Evaluating reward models for language modeling. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 1755–1797.

Lihua Lei and Emmanuel J Candès. 2021. Conformal inference of counterfactuals and individual treatment effects. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 83(5):911–938.

Dawei Li, Bohan Jiang, Liangjie Huang, Alimohammad Beigi, Chengshuai Zhao, Zhen Tan, Amrita Bhattacharjee, Yuxuan Jiang, Canyu Chen, Tianhao Wu, and 1 others. 2025a. From generation to judgment: Opportunities and challenges of llm-as-a-judge. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 2757–2791.

Junlong Li, Shichao Sun, Weizhe Yuan, Run-Ze Fan, Hai Zhao, and Pengfei Liu. 2023a. Generative judge for evaluating alignment. *arXiv preprint arXiv:2310.05470*.

Xinyu Li, Tianjin Huang, Ronghui Mu, Xiaowei Huang, and Gaojie Jin. 2025b. Pot: Inducing overthinking in llms via black-box iterative optimization. *arXiv preprint arXiv:2508.19277*.

Xinyu Li, Ronghui Mu, Lin Li, Tianjin Huang, and Gaojie Jin. 2026. OTora: A unified red teaming framework for reasoning-level denial-of-service in LLM agents. In *International Conference on Machine Learning*.

Zongjie Li, Chaozheng Wang, Pingchuan Ma, Daoyuan Wu, Shuai Wang, Cuiyun Gao, and Yang Liu. 2023b. Split and merge: Aligning position biases in large language model based evaluators.(2023). *arXiv preprint arXiv:2310.01432*.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2023. Let's verify step by step. In *The twelfth international conference on learning representations*.

Zhen Lin, Shubhendu Trivedi, and Jimeng Sun. 2023. Generating with confidence: Uncertainty quantification for black-box large language models. *arXiv preprint arXiv:2305.19187*.

Shuaitong Liu, Renjue Li, Lijia Yu, Lijun Zhang, Zhiming Liu, and Gaojie Jin. 2026. Badthink: Triggered overthinking attacks on chain-of-thought reasoning in large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-eval: Nlg evaluation using gpt-4 with better human alignment. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pages 2511–2522.

Adian Liusie, Vatsal Raina, Yassir Fathullah, and Mark Gales. 2024. Efficient llm comparative assessment: a product of experts framework for pairwise comparisons. In *Proceedings of the 2024 conference on empirical methods in natural language processing*, pages 6835–6855.

Potsawee Manakul, Adian Liusie, and Mark Gales. 2023. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pages 9004–9017.

Christopher Mohri and Tatsunori Hashimoto. 2024. Language models with conformal factuality guarantees. *arXiv preprint arXiv:2402.10978*.

Mahdi Pakdaman Naeini, Gregory Cooper, and Milos Hauskrecht. 2015. Obtaining well calibrated probabilities using bayesian binning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 29.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, and 1 others. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Arjun Panickssery, Samuel Bowman, and Shi Feng. 2024. Llm evaluators recognize and favor their own generations. *Advances in Neural Information Processing Systems*, 37:68772–68802.

Sangwoo Park, Matteo Zecchin, and Osvaldo Simeone. 2025. Adaptive prediction-powered autoeval with reliability and efficiency guarantees. *arXiv preprint arXiv:2505.18659*.

Robin L Plackett. 1975. The analysis of permutations. *Journal of the Royal Statistical Society Series C: Applied Statistics*, 24(2):193–202.

Victor Quach, Adam Fisch, Tal Schuster, Adam Yala, Jae Ho Sohn, Tommi S Jaakkola, and Regina Barzilay. 2023. Conformal language modeling. *arXiv preprint arXiv:2306.10193*.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741.

Yaniv Romano, Evan Patterson, and Emmanuel Candes. 2019. Conformalized quantile regression. *Advances in neural information processing systems*, 32.

Swarnadeep Saha, Omer Levy, Asli Celikyilmaz, Mohit Bansal, Jason Weston, and Xian Li. 2024. Branch-solve-merge improves large language model evaluation and generation. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 8352–8370.

Glenn Shafer and Vladimir Vovk. 2008. A tutorial on conformal prediction. *Journal of machine learning research*, 9(3).

Shreya Shankar, JD Zamfirescu-Pereira, Björn Hartmann, Aditya Parameswaran, and Ian Arawjo. 2024. Who validates the validators? aligning llm-assisted evaluation of llm outputs with human preferences. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*, pages 1–14.

Lin Shi, Chiyu Ma, Wenhua Liang, Xingjian Diao, Weicheng Ma, and Soroush Vosoughi. 2025. Judging the judges: A systematic study of position bias in llm-as-a-judge. In *Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*, pages 292–314.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. 2020. Learning to summarize with human feedback. *Advances in neural information processing systems*, 33:3008–3021.

Rickard Stureborg, Dimitris Alikaniotis, and Yoshi Suhara. 2024. Large language models are inconsistent and biased evaluators. *arXiv preprint arXiv:2405.01724*.

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

Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher D Manning. 2023. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5433–5442.

Pat Verga, Sebastian Hofstatter, Sophia Althammer, Yixuan Su, Aleksandra Piktus, Arkady Arkhangorodsky, Minjie Xu, Naomi White, and Patrick Lewis. 2024. Replacing judges with juries: Evaluating llm generations with a panel of diverse models. *arXiv preprint arXiv:2404.18796*.

Vladimir Vovk, Alexander Gammerman, and Glenn Shafer. 2005. *Algorithmic learning in a random world*. Springer.

Tu Vu, Kalpesh Krishna, Salaheddin Alzubi, Chris Tar, Manaal Faruqui, and Yun-Hsuan Sung. 2024. Foundational autoraters: Taming large language models for better automatic evaluation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 17086–17105.

Peiyi Wang, Lei Li, Liang Chen, Zefan Cai, Dawei Zhu, Binghuai Lin, Yunbo Cao, Lingpeng Kong, Qi Liu, Tianyu Liu, and 1 others. 2024. Large language models are not fair evaluators. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9440–9450.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Zixia Wang, Gaojie Jin, Jia Hu, and Ronghui Mu. 2026. Clucert: Certifying llm robustness via clustering-guided denoising smoothing. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Minghao Wu and Alham Fikri Aji. 2025. Style over substance: Evaluation biases for large language models. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 297–312.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. 2023. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms. *arXiv preprint arXiv:2306.13063*.

Yasin Abbasi Yadkori, Ilja Kuzborskij, David Stutz, András György, Adam Fisch, Arnaud Doucet, Iuliya Beloshapka, Wei-Hung Weng, Yao-Yuan Yang, Csaba Szepesvári, and 1 others. 2024. Mitigating llm hallucinations via conformal abstention. *arXiv preprint arXiv:2405.01563*.

Fanghua Ye, Mingming Yang, Jianhui Pang, Longyue Wang, Derek F Wong, Emine Yilmaz, Shuming Shi, and Zhaopeng Tu. 2024. Benchmarking llms via uncertainty quantification. *Advances in Neural Information Processing Systems*, 37:15356–15385.

Seonghyeon Ye, Doyoung Kim, Sungdong Kim, Hyeonbin Hwang, Seungone Kim, Yongrae Jo, James Thorne, Juho Kim, and Minjoon Seo. 2023. Flask: Fine-grained language model evaluation based on alignment skill sets. *arXiv preprint arXiv:2307.10928*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, and 1 others. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623.

Lianghui Zhu, Xinggang Wang, and Xinlong Wang. 2025. Judgelm: Fine-tuned large language models are scalable judges. In *International Conference on Learning Representations*, volume 2025, pages 51257–51296.

A Related Work

LLM-as-a-judge. LLM-based evaluation has emerged as a practical alternative to costly human annotation, powering automated benchmarks (Zheng et al., 2023; Dubois et al., 2023) and preference data collection at scale (Chiang and Lee, 2023). Subsequent work enriches the evaluation paradigm along multiple axes: chain-of-thought scoring (Liu et al., 2023), sub-criteria decomposition (Saha et al., 2024), and fine-grained skill-based assessment (Ye et al., 2023) improve prompting strategies; dedicated judge models such as Prometheus (Kim et al., 2024a,b), Auto-J (Li et al., 2023a), JudgeLM (Zhu et al., 2025), and FLAMe (Vu et al., 2024) improve alignment with human judgments through task-specific fine-tuning; and multi-judge approaches such as PoLL (Verga et al., 2024) and Product-of-Experts (Liusie et al., 2024) reduce single-model cost and bias. Despite these advances, LLM judges exhibit well-documented failure modes, including position bias (Zheng et al., 2023; Wang et al., 2024; Shi et al., 2025; Li et al., 2023b), self-preference bias (Panickssery et al., 2024; Stureborg et al., 2024), length and verbosity biases (Wu and Aji, 2025; Li et al., 2025a; Chen et al., 2024), with benchmark audits (Tan et al., 2024; Lambert et al., 2025; Shankar et al., 2024) further exposing reliability gaps. While the above methods improve accuracy or reduce biases empirically, none provides a formal guarantee on the resulting error rate. Furthermore, the vast majority of these studies operate in the pairwise regime (m=2), whereas practical pipelines such as best-of-N decoding in RLHF (Stiennon et al., 2020; Ouyang et al., 2022; Rafailov et al., 2023; Cobbe et al., 2021; Lightman et al., 2023) and multi-system comparisons require judging among m>2 candidates, a setting where probability dilution introduces additional reliability challenges.

Selective prediction and conformal risk control. This absence of formal guarantees motivates tools from selective prediction and conformal inference. Selective prediction (El-Yaniv et al., 2010; Geifman and El-Yaniv, 2017) trades coverage for reliability by allowing abstention under high uncertainty. Conformal prediction (Vovk et al., 2005; Shafer and Vovk, 2008; Angelopoulos and Bates, 2023; Romano et al., 2019; Lei and Candès, 2021) and distribution-free risk control (Bates et al., 2021; Angelopoulos et al., 2022, 2025; Gibbs and Candes, 2021) provide principled frameworks for calibrating such abstention thresholds with finite-sample guarantees. These tools have been adapted to LLM settings for calibrated abstention (Gui et al., 2024), token-level prediction sets (Quach et al., 2023), multi-choice QA (Kumar et al., 2023), hallucination mitigation (Yadkori et al., 2024; Manakul et al., 2023), and factuality guarantees (Mohri and Hashimoto, 2024), while complementary confidence-calibration techniques (Naeini et al., 2015; Guo et al., 2017; Kadavath et al., 2022; Wang et al., 2022; Xiong et al., 2023; Tian et al., 2023; Detommaso et al., 2024; Geng et al., 2024; Lin et al., 2023; Ye et al., 2024; Khanmohammadi et al., 2025) improve the underlying uncertainty estimates. While these methods provide formal guarantees for LLM generation and prediction tasks, they do not directly address the LLM-as-a-judge evaluation setting where the label space grows with m.

Guarantees for LLM evaluation. A separate line of work brings formal reliability guarantees into the LLM evaluation setting specifically. Most notably, Jung et al. (2024) propose Cascaded Selective Evaluation, which provides provable human-agreement guarantees for pairwise LLM judging (m=2) via fixed-sequence testing and introduces Simulated Annotators for confidence estimation. Concurrently, Badshah et al. (2026) propose SCOPE, which improves pairwise reliability through Bidirectional Preference Entropy (BPE), a bias-neutral uncertainty estimator that mitigates position bias. In a complementary direction, Park et al. (2025) study prediction-powered inference for automated evaluation with reliability guarantees. These works collectively demonstrate the growing need for principled reliability frameworks in LLM evaluation, yet all existing guarantee methods are restricted to the pairwise setting (m=2) (Jin et al., 2026).

Broader reliability landscape. Beyond evaluation-specific guarantees, research on trustworthy GenAI examines adversarially induced overthinking and reasoning-level denial-of-service attacks (Li et al., 2025b; Liu et al., 2026; Li et al., 2026), alongside systematic guardrail design (Dong et al., 2024) and certified LLM robustness (Wang et al., 2026). Complementary statistical work develops PAC-Bayesian generalization analyses based on weight correlation (Jin et al., 2020), second-order weight statistics (Jin et al.,

2022), and the spectral norm of robust confusion matrices (Jin et al., 2025).

Distinction from prior work. Our work extends the guarantee-based line of research from the pairwise to the multi-candidate regime (m>2). Unlike bias-mitigation methods (e.g., position swapping, multi-judge ensembling, fine-tuned judges) that reduce errors empirically without formal bounds, and unlike existing guarantee methods (Jung et al., 2024; Badshah et al., 2026) that are restricted to m=2, we identify a new structural challenge: the monotonic confidence–agreement relationship assumed by prior guarantee methods breaks down when probability mass is diluted across many alternatives. To address this, we propose a two-stage decomposition, conformal localization followed by selective auto-pick, that restores the monotonic structure and thereby enables valid guarantees for m>2. The framework is modular and orthogonal to the choice of scoring function: confidence estimators such as BPE (Badshah et al., 2026), Simulated Annotators (Jung et al., 2024), or any of the LLM-as-a-judge scoring methods reviewed above can serve as plug-in components within either stage.

B Simulation Details

The simulation in Figure 2 is designed to illustrate how the confidence–agreement monotonicity degrades as the number of candidates m grows, and how the proposed two-stage decomposition recovers it. We describe the generative model and the procedure used to produce each panel below.

Generative model. We construct a Plackett-Luce-inspired (Plackett, 1975) generative model that incorporates three realistic sources of variability: (i) heterogeneous instance difficulty, (ii) variable candidate quality gaps, and (iii) diverse annotator signal strengths. For each simulated instance with m candidates and N annotators, the data generation proceeds as follows.

1. **Instance difficulty.** Sample a difficulty parameter $d \sim \mathrm{Beta}(2, 5)$, which concentrates most instances on the "easy" end ($\mathbb{E}[d] = 0.29$) while occasionally producing difficult instances. The Gumbel noise scale is set to $\sigma = 0.5 + 1.5\,d$, so harder instances produce noisier annotator outputs.
2. **Candidate quality.** Sample the quality advantage of the ground-truth best candidate: $q_{\text{gap}} \sim \mathrm{Gamma}(2,\, 0.8)$. Assign quality $q_1 = q_{\text{gap}}$ to the true best response. For each remaining candidate $i \in \{2, \ldots, m\}$, independently with probability ρ=0.20, the candidate is competitive: $q_i = q_{\text{gap}} \cdot U(0.6, 1.0) - 0.2$; otherwise it is clearly inferior: $q_i \sim -\mathrm{Gamma}(1.5, 0.5)$. The competitive-candidate mechanism is a key driver of the monotonicity breakdown: at m=2 there are on average 0.2 competitive distractors per instance, whereas at m=20 there are ≈3.8, substantially increasing the frequency of high-confidence but incorrect predictions.
3. **Annotator probabilities.** For each annotator $j \in \{1, \ldots, N\}$, sample a signal strength $s_j \sim \mathrm{Beta}(3, 2)$ (mean 0.6; strong annotators approach 0.9, weak ones fall to ≈0.2). The perceived quality of candidate i by annotator j is

$$\tilde{q}_{j,i} \;=\; q_i{\cdot}s_j + \epsilon_{j,i}, \qquad \epsilon_{j,i} \sim \mathrm{Gumbel}(0,\, \sigma),$$

   and the annotator's probability vector is $p_{j,i} = \mathrm{softmax}(\tilde{q}_{j,1}, \ldots, \tilde{q}_{j,m})_i$. The multiplicative signal attenuation $q_i \cdot s_j$ compresses the quality gap for weak annotators, making their outputs closer to uniform and thus less informative—analogous to poorly chosen few-shot prompts in real Simulated Annotators.
4. **Position debiasing.** Candidate indices are randomly permuted to remove any positional artifact. The ensemble mean probability (EMP) is then computed as $\bar{p}_i = \frac{1}{N}\sum_{j=1}^{N} p_{j,i}$.

Why this model captures the real phenomenon. The interaction between the number of candidates m and the competitive-candidate mechanism creates a realistic probability dilution effect. As m increases, more competitive distractors appear (on average $\rho(m{-}1)$), splitting the EMP probability mass among confusable alternatives and shrinking the numerical margin between the best and second-best candidates. This produces the non-monotonic confidence–agreement patterns observed in Figure 2(a) and motivates the two-stage solution shown in Figure 2(b)–(c).

Panel descriptions. In all three panels of Figure 2, each point at confidence threshold λ reports the average agreement rate over all instances whose estimated confidence satisfies $c(x) \geq \lambda$. Monotonicity means that this agreement rate is non-decreasing in λ: a stricter threshold should yield higher accuracy among accepted instances.

- **Panel (a)** (Confidence–Agreement Monotonicity): For each $m \in \{2, 5, 10, 20\}$, the confidence

score is $c(x) = \max_i \bar{p}_i$ and the agreement indicator is $\mathbf{1}[\arg\max_i \bar{p}_i = y]$. As m increases, the confidence range compresses (the maximum EMP score decreases due to probability dilution among more candidates) and the monotonic relationship between confidence and agreement progressively degrades.

- **Panel (b)** (Top-k Localization): Fixing m=20, we compare direct top-1 selection (k=1) against top-k localization (k=3). The confidence score remains $c(x) = \max_i \bar{p}_i$; the agreement indicator changes to $\mathbf{1}[y \in \text{top-}k(\bar{p})]$, i.e., whether the ground-truth best lies within the top-k candidates ranked by EMP. While top-1 agreement is non-monotonic, top-3 agreement recovers a clean monotonic curve with the same confidence score, motivating Stage I conformal localization.
- **Panel (c)** (Monotonicity Within the Shortlist): Fixing m=20 and restricting to instances where y lies in the top-3 shortlist $\mathcal{S}$, we compare four confidence functions defined in Appendix E: EMP top-1, KL top-1, EMP margin, and KL margin. The agreement indicator is $\mathbf{1}[\hat{y} = y \mid y \in \mathcal{S}]$ where $\hat{y}$ is the predicted best within $\mathcal{S}$. Margin-based confidence functions (especially KL margin) produce substantially smoother and more monotonic reliability curves than top-1 confidence, motivating the margin-based Stage II design adopted in our framework.

Configuration. Simulations use N=25 annotators and 20,000 instances per setting, with agreement rates evaluated at evenly spaced confidence thresholds. Table 4 summarizes the complete parameter settings.

Parameter	Value	Description
m	$\{2, 5, 10, 20\}$	Candidates per instance
N	25	Simulated annotators
Instances	20,000	Per m setting
q_{gap}	$\text{Gamma}(2, 0.8)$	Best-candidate advantage
Comp. prob. ρ	0.20	Competitive distractor
Non-best q_i	$-\text{Gamma}(1.5, 0.5)$	Weak-candidate quality
Difficulty d	$\text{Beta}(2, 5)$	Gumbel noise scale
Signal s_j	$\text{Beta}(3, 2)$	Annotator strength
Noise ϵ	$\text{Gumbel}(0, 0.5{+}1.5d)$	Perception noise

Table 4: Simulation configuration for Figure 2.

C Proofs

Proof 1 (Proof of Theorem 3.1) *Let $z_1, \ldots, z_n, z_{n+1}$ be exchangeable and define $R_i = R(z_i)$ for $i = 1, \ldots, n+1$. By exchangeability, the joint law of $(R_1, \ldots, R_n, R_{n+1})$ is invariant under permutations. Therefore, the (tie-broken) rank of R_{n+1} among $\{R_1, \ldots, R_n, R_{n+1}\}$ is uniform on $\{1, \ldots, n+1\}$. Let k be defined by (2). The standard conformal quantile argument yields $\mathbb{P}(R_{n+1} \le k) \ge 1 - \alpha$. That means $R(z) \le k$ holds if and only if the human-best index lies in the top-k candidates under $\mathbb{S}_c(x, \cdot)$, i.e., $y \in \mathcal{S}(x)$. Hence (4) holds. The statement $|\mathcal{S}(x)| = k$ follows by construction.* □

Proof 2 (Proof of Theorem 3.2) *Fix $\varepsilon \in (0, 1)$ and define*

$$R^{\text{in}}_{\mathcal{S}}(\lambda) = \mathbb{P}(\hat{y} \neq y \mid \Delta_{\mathcal{S}}(x) \ge \lambda,\ y \in \mathcal{S}(x)).$$

For each λ, let $N_{\text{in}}(\lambda)$ be the number of calibration samples in $\mathcal{D}_{\text{in}} = \{(x_i, y_i) \in \mathcal{D} :\ y_i \in \mathcal{S}(x_i)\}$ satisfying $\Delta_{\mathcal{S}}(x_i) \ge \lambda$, and let $Y_{\text{in}}(\lambda)$ be the number of errors among them. Conditional on $N_{\text{in}}(\lambda)$,

$$Y_{\text{in}}(\lambda) \sim \text{Bin}\big(N_{\text{in}}(\lambda), R^{\text{in}}_{\mathcal{S}}(\lambda)\big).$$

By exact binomial inversion, the upper confidence bound in (8) satisfies, for each fixed λ,

$$\mathbb{P}\Big(R^{\text{in}}_{\mathcal{S}}(\lambda) > \widehat{R}^{\text{in},+}_{\mathcal{S}}(\lambda)\Big) \le \delta.$$

Applying the fixed-sequence testing rule, we test from the largest value of λ (e.g., 0.999) to a progressively smaller value, and stop at the last time $R^{\text{in},+}_{\mathcal{S}}(\lambda)$ is below the target risk ε, then we get last time λ^*. Since $\widehat{R}^{\text{in},+}_{\mathcal{S}}(\lambda^*) \le \varepsilon$ is equivalent to rejecting $H_\lambda : R^{\text{in}}_{\mathcal{S}}(\lambda^*) > \varepsilon$ at level δ, the fixed-sequence testing guarantee implies that the probability of rejecting any true null hypothesis is at most δ, and therefore

$$\mathbb{P}\big(R^{\text{in}}_{\mathcal{S}}(\lambda^*) > \varepsilon\big) \le \delta.$$

Therefore, with probability at least $1 - \delta$ over the calibration sample,

$$\mathbb{P}(\hat{y} \neq y \mid \Delta_{\mathcal{S}}(x) \ge \lambda^*,\ y \in \mathcal{S}(x)) \le \varepsilon,$$

which completes the proof. □

Proof 3 (Proof of Corollary 3.3) *Let $A := \{\Delta_{\mathcal{S}}(x) \ge \lambda^*\}$. By the law of total probability,*

$$\begin{aligned}\mathbb{P}(\hat{y} = y \mid A) &= \mathbb{P}(\hat{y} = y,\ y \in \mathcal{S}(x) \mid A)\\ &= \mathbb{P}(\hat{y} = y \mid A,\ y \in \mathcal{S}(x))\\ &\quad \cdot \mathbb{P}(y \in \mathcal{S}(x) \mid A).\end{aligned}$$

By Theorem 3.2, with probability at least $1-\delta$ over the calibration sample used in Stage II,

$$\mathbb{P}(\hat{y} \neq y \mid A,\, y \in \mathcal{S}(x)) \leq \varepsilon,$$

hence $\mathbb{P}(\hat{y} = y \mid A,\, y \in \mathcal{S}(x)) \geq 1-\varepsilon$.

It remains to lower bound $\mathbb{P}(y \in \mathcal{S}(x) \mid A)$. By the monotonicity assumption that $\lambda \mapsto \mathbb{P}(y \notin \mathcal{S}(x) \mid \Delta_{\mathcal{S}}(x) \geq \lambda)$ is nonincreasing for $\lambda \geq 0$, we have

$$\mathbb{P}(y \notin \mathcal{S}(x) \mid A) \leq \mathbb{P}(y \notin \mathcal{S}(x) \mid \Delta_{\mathcal{S}}(x) \geq 0).$$

Since $\Delta_{\mathcal{S}}(x) \geq 0$ almost surely in our setting, the event $\{\Delta_{\mathcal{S}}(x) \geq 0\}$ is the entire sample space, and thus

$$\mathbb{P}(y \notin \mathcal{S}(x) \mid \Delta_{\mathcal{S}}(x) \geq 0) = \mathbb{P}(y \notin \mathcal{S}(x)).$$

By Theorem 3.1, $\mathbb{P}(y \notin \mathcal{S}(x)) \leq \alpha$, hence

$$\mathbb{P}(y \in \mathcal{S}(x) \mid A) = 1 - \mathbb{P}(y \notin \mathcal{S}(x) \mid A) \geq 1-\alpha.$$

Combining the two bounds yields, with probability at least $1-\delta$,

$$\begin{aligned}\mathbb{P}(\hat{y} = y \mid \Delta_{\mathcal{S}}(x) \geq \lambda^*) \\ = \mathbb{P}(\hat{y} = y \mid A) \geq (1-\varepsilon)(1-\alpha),\end{aligned}$$

which proves the corollary. □

D Multi-Candidate Data Construction

Each of our four evaluation benchmarks originally provides pairwise human-preference annotations. To construct multi-candidate instances with $m \in \{5, 10, 20\}$ candidates per query, we follow a general three-step procedure.

Candidate pool construction. For each dataset, we group all pairwise-comparison records by query and collect every distinct candidate response, forming a base candidate pool. We retain the top 3,000 queries ranked by the number of available distinct candidates. When the base pool for a query contains fewer candidates than the largest target m, we augment it with additional responses generated by a separate set of open-weight LLMs that do not overlap with the judge models used in our main experiments, so as to avoid any interaction between candidate generation and evaluation. To prevent the generated candidates from exceeding the quality of the human-preferred response, each generation prompt uses a non-preferred candidate from the base pool as a stylistic reference rather than the ground-truth best response. After generation, exact-string deduplication is applied to ensure all candidates in the pool are distinct.

Ground-truth assignment. For each query, the ground-truth preferred response is identified as the candidate most frequently preferred by human annotators across all available pairwise comparisons.

Instance sampling. For each target m, we include the ground-truth preferred response together with $m-1$ randomly sampled alternatives from the candidate pool, shuffle the ordering, and record the position of the preferred response as the label y. When the candidate pool is sufficiently large relative to m, multiple distinct m-candidate instances can be drawn from the same query by sampling different candidate subsets. This yields approximately 3,000 instances per (dataset, m) pair across all four benchmarks. All judge models are evaluated on the identical set of instances per dataset, enabling post-hoc composition of cascade configurations.

E Scoring Functions

This section defines all scoring functions used in our experiments. All scores are computed from the stored Simulated Annotator probability vectors (Jung et al., 2024): for each instance x with m candidates, we collect $\mathbf{p}_j = (p_{j,1}, \ldots, p_{j,m})$ from each of $N{=}5$ annotators (each conditioned on $K{=}5$ few-shot examples), where $p_{j,i} = P_\theta(y{=}i \mid x; \text{annotator}_j)$.

E.1 Stage I: Conformal Localization Scoring

Stage I uses a per-candidate score $\mathbb{S}_c(x, g_i)$ to rank all m candidates for conformal localization. Throughout this paper we use the EMP:

$$\mathbb{S}_c(x, g_i) \;=\; \bar{p}_i \;=\; \frac{1}{N}\sum_{j=1}^{N} p_{j,i},$$

the average predicted probability across annotators. The top-k candidates under EMP form the shortlist $\mathcal{S}(x)$. This scoring function is used exclusively for Stage I localization and is fixed throughout all experiments.

E.2 Stage II and Single-Stage: Selection Scoring

Stage II of our two-stage framework and the single-stage baseline of Jung et al. (2024) share the same set of scoring functions; the only difference is the candidate set over which they operate: Stage II scores candidates within the localized shortlist $\mathcal{S}(x)$, while the single-stage baseline scores all m candidates directly. In both cases, the system

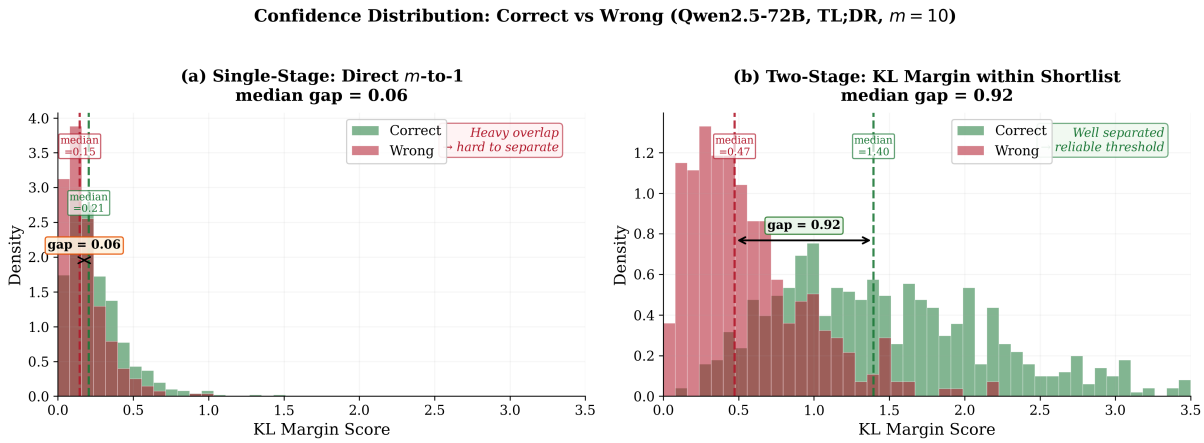

Figure 5: **Confidence distributions for correct vs. wrong predictions** (Qwen2.5-72B, TL;DR, m=10). **(a)** Single-stage direct m-to-1 KL margin: the correct and wrong distributions overlap heavily (median gap $= 0.06$), making threshold-based selection unreliable. **(b)** Two-stage KL margin within the shortlist: after Stage I localization, the distributions are well separated (median gap $= 0.92$), confirming that localization improves the discriminative power of Stage II scoring.

Dataset	Judge	Direct m-to-1			Stage I (m-to-k)			Stage II (k-to-1)		
		m=5	m=10	m=20	m=5	m=10	m=20	m=5	m=10	m=20
TL;DR	Mistral-7B	0.22	0.33	0.44	0.03	0.05	0.06	0.02	0.03	0.05
	Qwen2.5-32B	0.17	0.28	0.38	0.02	0.03	0.05	0.01	0.02	0.03
	Llama-3-70B	0.15	0.25	0.35	0.02	0.03	0.04	0.01	0.02	0.03
Chatbot Arena	Mistral-7B	0.25	0.36	0.47	0.04	0.06	0.08	0.03	0.05	0.06
	Qwen2.5-32B	0.19	0.30	0.42	0.03	0.04	0.06	0.02	0.03	0.05
	Llama-3-70B	0.16	0.27	0.38	0.02	0.03	0.05	0.01	0.02	0.04
HH-RLHF	Mistral-7B	0.20	0.31	0.41	0.03	0.05	0.07	0.02	0.04	0.05
	Qwen2.5-32B	0.15	0.25	0.36	0.02	0.03	0.05	0.01	0.02	0.04
	Llama-3-70B	0.14	0.22	0.32	0.01	0.02	0.04	0.01	0.01	0.03
AlpacaEval	Mistral-7B	0.23	0.35	0.46	0.04	0.06	0.07	0.03	0.04	0.06
	Qwen2.5-32B	0.18	0.29	0.40	0.02	0.04	0.05	0.01	0.03	0.04
	Llama-3-70B	0.15	0.24	0.36	0.02	0.03	0.04	0.01	0.02	0.03

Table 5: **Ranking loss (additional judges)** (T=20 thresholds, α=0.10). Same format as Table 1.

uses a per-candidate scoring function $\mathbb{S}_h$ to rank candidates, then applies a confidence function to decide whether to accept or abstain.

Per-candidate scoring functions. Given a candidate set C (the shortlist $\mathcal{S}(x)$ for Stage II, or the full set $\{1, \ldots, m\}$ for the single-stage baseline), a per-candidate scoring function assigns a score $\mathbb{S}_h(x, g_i)$ to each $i \in C$. The predicted winner is $\hat{y} = \arg\max_{i \in C} \mathbb{S}_h(x, g_i)$. We consider three choices:

- **EMP.** $\mathbb{S}_h(x, g_i) = \bar{p}_i = \frac{1}{N}\sum_{j=1}^{N} p_{j,i}$. This uses the same ensemble mean probability as Stage I but serves a different role: here it provides per-candidate scores for the selection decision within C, rather than for conformal localization over all m candidates.
- **KL-divergence scoring with variance normalization.** We derive the per-candidate score from KL divergence. First, we renormalize each annotator's probabilities to the candidate set C: $q_i^{(j)} = p_{j,i} / \sum_{l \in C} p_{j,l}$, which yields a valid distribution over C and removes the influence of eliminated candidates. To measure how strongly annotator j supports candidate i, we compute the KL divergence from the point mass e_i (i.e., the distribution that places all mass on i) to the annotator's renormalized distribution: $\mathrm{KL}(e_i \| \mathbf{q}^{(j)}) = -\log q_i^{(j)}$. A smaller KL divergence indicates stronger support. Since KL is a distance (smaller is better) but we need a weight (larger is better), we apply the exponential transform: $\exp(-\mathrm{KL}(e_i \| \mathbf{q}^{(j)})) = q_i^{(j)}$. That is, the renormalized probability $q_i^{(j)}$ directly serves as the KL-derived voting weight. The per-candidate score aggregates this weight across annotators who vote for i and normalizes

Algorithm 1 Two-Stage Certified Selection (Ours)

Require: Calibration set $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^n$; test input x with m candidates; risk levels $\alpha, \varepsilon, \delta$; threshold grid $\Lambda = \{0.99, \cdots, 0.00\}$
Require: Stage I scoring $\mathbb{S}_c$ *(default: EMP)*; Stage II scoring $\mathbb{S}_h$ *(plug-in: EMP, KL, or Vote)*; confidence type *(top-1 or margin)*

Stage I: shortlist size k

for $i = 1, \ldots, n$ **do**
  $R(z_i) \leftarrow |\{j : \mathbb{S}_c(x_i, g_j) \geq \mathbb{S}_c(x_i, g_{y_i})\}|$ ▷ rank of true label (1)
end for
$k \leftarrow \lceil (n+1)(1-\alpha) \rceil$-th order statistic of $\{R(z_i)\}_{i=1}^n$ (2)

Stage II: threshold λ^*

$\mathcal{S}(x_i) \leftarrow$ top-k under $\mathbb{S}_c(x_i, \cdot), \forall i$ (3)
$\mathcal{D}_{\text{in}} \leftarrow \{(x_i, y_i) \in \mathcal{D} : y_i \in \mathcal{S}(x_i)\}$
for $j \in \mathcal{D}_{\text{in}}$ **do**
  $\hat{y}_j \leftarrow \arg\max_{i \in \mathcal{S}(x_j)} \mathbb{S}_h(x_j, g_i)$
  $c_j \leftarrow \textsc{Conf}(\hat{y}_j, \mathcal{S}(x_j), \mathbb{S}_h)$ Alg. 3
end for
$\lambda^* \leftarrow -\infty$
for $\lambda \in \Lambda$ **(largest $\rightarrow$ smallest) do**
  $N^{\text{in}} \leftarrow |\{j \in \mathcal{D}_{\text{in}} : c_j \geq \lambda\}|$
  $Y^{\text{in}} \leftarrow \sum_{j \in \mathcal{D}_{\text{in}},\, c_j \geq \lambda} \mathbf{1}[\hat{y}_j \neq y_j]$
  $\widehat{R}^{\text{in},+} \leftarrow \text{BinomUCB}_\delta(Y^{\text{in}}, N^{\text{in}})$ (8)
  if $\widehat{R}^{\text{in},+} \leq \varepsilon$ **then** $\lambda^* \leftarrow \lambda$; **else break** (9)
end for

Inference (new x)

$\mathcal{S}(x) \leftarrow$ indices of top-k candidates under $\mathbb{S}_c(x, \cdot)$ ▷ Stage I: localize
$\hat{y} \leftarrow \arg\max_{i \in \mathcal{S}(x)} \mathbb{S}_h(x, g_i)$ ▷ Stage II: predict
$c(x) \leftarrow \textsc{Conf}(\hat{y}, \mathcal{S}(x), \mathbb{S}_h)$ Alg. 3
return $\hat{y}$ if $c(x) \geq \lambda^*$; ABSTAIN O.W.

Algorithm 2 Single-Stage Baseline (Jung et al., 2024)

Require: Calibration set $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^n$; test input x with m candidates; risk level ε', δ; threshold grid $\Lambda = \{0.99, \cdots, 0.00\}$
Require: Per-candidate scoring $\mathbb{S}_h$ *(plug-in: EMP, KL, or Vote)*; confidence type *(top-1 or margin)*

Calibration (no localization)

$C \leftarrow \{1, \ldots, m\}$
for $j \in \mathcal{D}$ **do**
  $\hat{y}_j \leftarrow \arg\max_{i \in C} \mathbb{S}_h(x_j, g_i)$
  $c_j \leftarrow \textsc{Conf}(\hat{y}_j, C, \mathbb{S}_h)$ Alg. 3
end for
$\lambda^* \leftarrow -\infty$
for $\lambda \in \Lambda$ **(largest $\rightarrow$ smallest) do**
  $N \leftarrow |\{j : c_j \geq \lambda\}|$
  $Y \leftarrow \sum_{j:\, c_j \geq \lambda} \mathbf{1}[\hat{y}_j \neq y_j]$
  $\widehat{R}^+ \leftarrow \text{BinomUCB}_\delta(Y, N)$
  if $\widehat{R}^+ \leq \varepsilon'$ **then** $\lambda^* \leftarrow \lambda$; **else break**
end for

Inference (new x)

$\hat{y} \leftarrow \arg\max_{i \in C} \mathbb{S}_h(x, g_i)$
$c(x) \leftarrow \textsc{Conf}(\hat{y}, C, \mathbb{S}_h)$
return $\hat{y}$ if $c(x) \geq \lambda^*$; ABSTAIN O.W.

Algorithm 3 CONF: Confidence Function

Require: Winner $\hat{y}$; candidate set C; scoring $\mathbb{S}_h$
$s_1 \leftarrow \mathbb{S}_h(x, g_{\hat{y}})$
$s_2 \leftarrow \max_{i \in C,\, i \neq \hat{y}} \mathbb{S}_h(x, g_i)$
if top-1 mode **then**
  return s_1
else if margin mode **then** (5)
  return $s_1 - s_2$
end if

for cross-instance comparability:

$$\mathbb{S}_h(x, g_i) = \frac{\frac{1}{N}\sum_{j=1}^N q_i^{(j)} \cdot \mathbf{1}\{\arg\max_{l \in C} q_l^{(j)} = i\}}{\sqrt{\sigma_C^2 + \bar{v}_i}}, \tag{11}$$

where the numerator averages the KL-derived weight $q_i^{(j)}$ over the annotators whose top choice within C is i; $\bar{v}_i$ is the variance of $q_i^{(j)}$ among those voters (rewarding consensus); and σ_C^2 is the variance of $\{\bar{q}_i\}_{i \in C}$ across candidates (a shared scale factor that enables cross-instance comparability and prevents the denominator from collapsing when $\bar{v}_i$ is small).

- **Vote.** The vote share for candidate i: $\mathbb{S}_h(x, g_i) = \frac{1}{N}\sum_{j=1}^N \mathbf{1}\big[\arg\max_{i' \in C} p_{j,i'} = i\big]$, i.e., the fraction of annotators whose top-1 prediction within C is i. This is a non-parametric alternative that discards probability magnitudes and only counts consensus.

Confidence functions. Given per-candidate scores $\mathbb{S}_h(x, g_i)$ and the predicted winner $\hat{y}$, a confidence function produces a scalar that is compared against the calibrated threshold λ^* to decide acceptance. We consider two types:

- **Top-1.** The confidence is the score of the predicted winner: $c(x) = \mathbb{S}_h(x, g_{\hat{y}})$. The system accepts if $c(x) \geq \lambda^*$. This captures whether the top candidate has a sufficiently high absolute score.
- **Margin.** The confidence is the gap between the top two candidates: $c(x) = \mathbb{S}_h(x, g_{\hat{y}}) - \max_{i \in C,\, i \neq \hat{y}} \mathbb{S}_h(x, g_i)$. The system accepts if $c(x) \geq \lambda^*$. This captures whether the top candidate is sufficiently separated from the runner-up, corresponding to $\Delta_{\mathcal{S}}(x)$ in (5).

Named configurations. Combining three per-candidate scoring functions with two confidence functions yields the following named configurations, used consistently in both the single-stage and two-stage columns of Tables 2 and 6:

Name in tables	Per-candidate scoring	Confidence type
top-1 EMP	EMP	top-1
EMP margin	EMP	margin
top-1 KL	KL-divergence scoring	top-1
KL margin	KL-divergence scoring	margin
Vote	Vote	top-1

F Algorithm Pseudocode

Algorithm 1 presents the full pseudocode for our two-stage certified selection framework. The scoring functions $\mathbb{S}_c$ (Stage I) and $\mathbb{S}_h$ (Stage II) are plug-in components: $\mathbb{S}_c$ is fixed to EMP throughout, while $\mathbb{S}_h$ can be any of the per-candidate scoring functions defined in Appendix E.2 (EMP, KL, or Vote), combined with either a top-1 or margin confidence function. Algorithm 2 shows the single-stage baseline of Jung et al. (2024), which corresponds to applying Stage II directly to all m candidates without prior localization. Both algorithms are structured in two phases: calibration (offline, on $\mathcal{D}$) and test-time inference (online, per new x). The key structural difference is that Algorithm 1 first determines the shortlist size k via conformal calibration and then calibrates λ^* only on the covered subset $\mathcal{D}_{\text{in}}$, whereas Algorithm 2 calibrates λ^* directly on all m candidates without localization.

G Monotonicity Evaluation Settings and Confidence Distribution Analysis

This appendix provides the detailed experimental settings used to produce the monotonicity results in Table 1 and Figure 3, and presents additional confidence-distribution analysis.

Confidence and agreement definitions. Table 1 and Figure 3 share the same three evaluation settings, each corresponding to a distinct stage of the selection pipeline. The confidence and agreement definitions for each setting are as follows:

- **Direct m-to-1 (single-stage baseline):** The confidence score is the top-1 EMP, i.e., $c(x) = \max_i \bar{p}_i$ where $\bar{p}_i = \frac{1}{N}\sum_{j=1}^{N} p_{j,i}$. The agreement indicator is $\mathbf{1}[\hat{y} = y]$, where $\hat{y} = \arg\max_i \bar{p}_i$.
- **Stage I (m-to-k localization):** The confidence score is the same top-1 EMP as above. The agreement indicator changes to $\mathbf{1}[y \in \text{top-}k]$, i.e., whether the human-preferred response lies within the top-k candidates ranked by EMP. The shortlist size k is determined by the conformal quantile (Equation (2)) with α=0.10.
- **Stage II (k-to-1 selection within shortlist):** The confidence score is the KL-divergence margin $\Delta_{\mathcal{S}}(x)$ defined in Equation (5), computed within the shortlist $\mathcal{S}(x)$. The agreement indicator is $\mathbf{1}[\hat{y} = y \mid y \in \mathcal{S}(x)]$, restricted to instances where the human-preferred response is covered by the shortlist.

Ranking loss computation (Table 1). For Table 1, the T=20 confidence thresholds are evenly spaced over the observed range $[\min_i c(x_i), \max_i c(x_i)]$ for each (dataset, judge, m) combination. The ranking loss (defined in Section 4.2) is computed on the full dataset without calibration/test splitting to maximize statistical stability.

Confidence–agreement curves (Figure 3). Figure 3 visualizes the confidence–agreement relationship using Qwen2.5-72B on TL;DR as a representative pair, chosen because it is a mid-strength judge that clearly exhibits the monotonicity breakdown in single-stage evaluation. The three panels correspond to the three settings defined above: panel (a) sweeps $m \in \{2, 5, 10, 20\}$ under the Direct m-to-1 setting; panel (b) fixes m=20 and compares k=1 (direct selection) against k=3 (localization); panel (c) fixes m=20 within the top-3 shortlist and compares four Stage II scoring functions (top-1 EMP, top-1 KL, EMP margin, and KL margin, as defined in Appendix E). The same qualitative patterns hold for other (dataset, judge) combinations, as confirmed by Tables 1 and 5. Figures 6 and 7 present the corresponding figures for Chatbot Arena (Qwen2.5-72B) and HH-RLHF (GPT-OSS-120B), showing that the monotonicity breakdown and two-stage recovery are consistent across both datasets and judge models.

Confidence-score distributions (Figure 5). Figure 5 visualizes the distributions of KL margin scores for correct versus incorrect predictions. In the single-stage setting (panel a), the KL margin is computed over all m candidates; in the two-stage setting (panel b), it is computed within the shortlist $\mathcal{S}(x)$. After Stage I localization, the correct and incorrect distributions become substantially more separated (median gap increases from 0.06 to 0.92), confirming that localization improves the discriminative power of Stage II scoring and enables more reliable threshold-based selection.

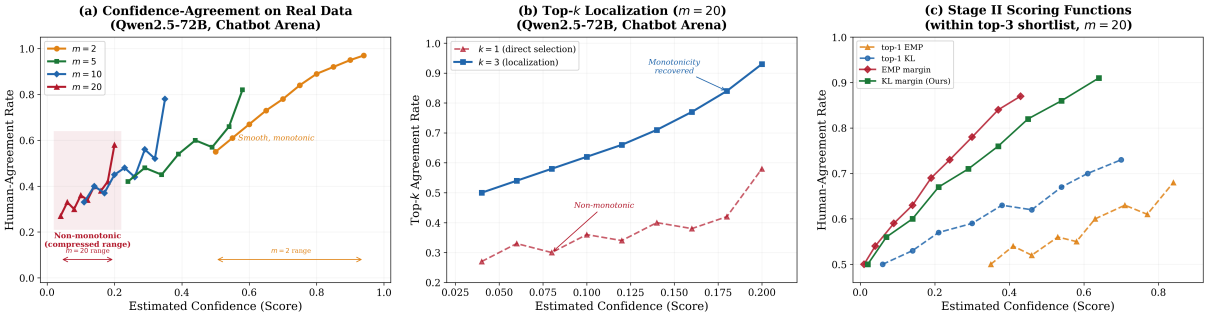

Figure 6: **Confidence–agreement curves on Chatbot Arena and Qwen2.5-72B.** Same layout as Figure 3. **(a)** Single-stage m-to-1. **(b)** Stage I top-k localization. **(c)** Stage II scoring within the top-3 shortlist (m=20).

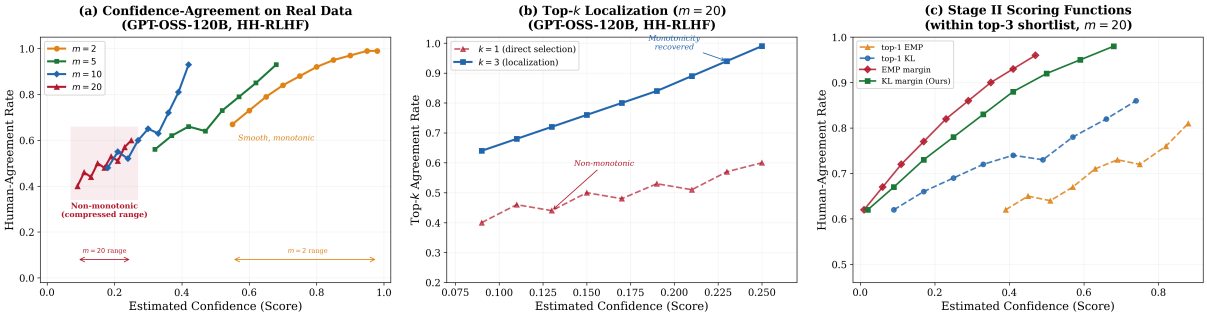

Figure 7: **Confidence–agreement curves on HH-RLHF and GPT-OSS-120B.** Same layout as Figure 3. **(a)** Single-stage m-to-1. **(b)** Stage I top-k localization. **(c)** Stage II scoring within the top-3 shortlist (m=20).

H Additional Validation and Ablation Studies

This section reports targeted analyses of the cross-stage monotonicity condition, a strengthened single-stage baseline, and the components of the KL-margin score.

H.1 Direct Validation of Cross-Stage Monotonicity

For each configuration, we estimate $\mathbb{P}(y \notin \mathcal{S}(x) \mid \Delta_{\mathcal{S}}(x) \geq \lambda)$ over 1,000 random calibration/test splits with α=0.10. Table 8 reports representative configurations.

For every configuration shown, the conditional miss probability decreases monotonically with λ. At λ=0, it remains below the marginal failure level α=0.10; at λ=0.8, it falls below 1%.

Interpreting GSR. For each split, empirical agreement on accepted test instances estimates the inner population quantity $\mathbb{P}_{(x,y)}(\hat{y} = y \mid \Delta_{\mathcal{S}}(x) \geq \lambda^*)$. Across the 1,000 random calibration splits, GSR estimates the frequency with which this empirical agreement reaches the target $(1-\varepsilon)(1-\alpha)$, corresponding to the outer confidence requirement $1-\delta$. Because each test set is finite, GSR should be interpreted as an empirical validation of the nested guarantee rather than an exact observation of the population probability.

H.2 Variance-Normalized Single-Stage Baseline

To test whether confidence-level normalization alone resolves probability dilution, we augment the direct m-to-1 adaptation of Jung et al. (2024) with variance normalization. Table 9 compares this strengthened baseline with the strongest single-stage scoring baseline and our two-stage method under the same representative setting.

In this setting, variance normalization does not close the gap: the two-stage method improves both GSR and coverage over the strengthened single-stage alternatives.

H.3 Component-Wise Ablation of KL Margin

We ablate the five components of the KL-margin score in (11), removing one component at a time while retaining the other four.

In this representative setting, removing any component reduces both GSR and coverage. Shortlist renormalization and variance normalization produce the largest drops, while the remaining components provide smaller complementary gains.

Judge	Method	TL;DR			Chatbot Arena			HH-RLHF			AlpacaEval		
		Cov.	GSR	Agr.	Cov.	GSR	Agr.	Cov.	GSR	Agr.	Cov.	GSR	Agr.
Mistral 7B	No Selection	100%	0%	$35\%_{\pm 1.3}$	100%	0%	$30\%_{\pm 1.5}$	100%	0%	$39\%_{\pm 1.1}$	100%	0%	$33\%_{\pm 1.4}$
	Single-Stage (EMP)	$25\%_{\pm 3.3}$	52.4%	$77\%_{\pm 2.3}$	$20\%_{\pm 3.7}$	46.8%	$74\%_{\pm 2.5}$	$29\%_{\pm 3.0}$	57.6%	$79\%_{\pm 2.0}$	$23\%_{\pm 3.5}$	50.3%	$76\%_{\pm 2.3}$
	Single-Stage (EMP margin)	$30\%_{\pm 3.1}$	57.3%	$78\%_{\pm 2.1}$	$25\%_{\pm 3.4}$	51.8%	$76\%_{\pm 2.3}$	$34\%_{\pm 2.8}$	62.7%	$80\%_{\pm 1.9}$	$28\%_{\pm 3.2}$	55.1%	$77\%_{\pm 2.2}$
	Single-Stage (KL margin)	$34\%_{\pm 2.9}$	62.4%	$80\%_{\pm 2.0}$	$29\%_{\pm 3.2}$	56.9%	$78\%_{\pm 2.1}$	$38\%_{\pm 2.6}$	67.8%	$82\%_{\pm 1.8}$	$32\%_{\pm 3.0}$	60.6%	$79\%_{\pm 2.1}$
	Single-Stage (Vote)	$32\%_{\pm 3.0}$	59.8%	$79\%_{\pm 2.0}$	$27\%_{\pm 3.3}$	54.1%	$77\%_{\pm 2.2}$	$36\%_{\pm 2.7}$	64.5%	$81\%_{\pm 1.8}$	$30\%_{\pm 3.1}$	57.9%	$78\%_{\pm 2.1}$
	Two-Stage (EMP margin)	$38\%_{\pm 2.7}$	**90.6%**	$84\%_{\pm 1.6}$	$33\%_{\pm 3.0}$	**90.1%**	$82\%_{\pm 1.8}$	$42\%_{\pm 2.4}$	**92.3%**	$85\%_{\pm 1.4}$	$36\%_{\pm 2.8}$	**90.4%**	$83\%_{\pm 1.7}$
	Two-Stage (KL margin)	$44\%_{\pm 2.4}$	**94.7%**	$86\%_{\pm 1.4}$	$39\%_{\pm 2.7}$	**93.2%**	$85\%_{\pm 1.6}$	$48\%_{\pm 2.2}$	**95.6%**	$87\%_{\pm 1.2}$	$42\%_{\pm 2.5}$	**93.8%**	$86\%_{\pm 1.5}$
	Two-Stage (Vote)	$41\%_{\pm 2.5}$	**92.8%**	$85\%_{\pm 1.5}$	$36\%_{\pm 2.8}$	**91.4%**	$83\%_{\pm 1.7}$	$45\%_{\pm 2.3}$	**93.7%**	$86\%_{\pm 1.3}$	$39\%_{\pm 2.6}$	**92.1%**	$85\%_{\pm 1.6}$
Qwen2.5 32B	No Selection	100%	0%	$44\%_{\pm 1.2}$	100%	0%	$39\%_{\pm 1.3}$	100%	0%	$47\%_{\pm 1.0}$	100%	0%	$43\%_{\pm 1.2}$
	Single-Stage (EMP)	$34\%_{\pm 2.8}$	57.6%	$79\%_{\pm 2.0}$	$29\%_{\pm 3.2}$	52.3%	$77\%_{\pm 2.2}$	$38\%_{\pm 2.6}$	61.8%	$81\%_{\pm 1.8}$	$33\%_{\pm 2.9}$	56.4%	$78\%_{\pm 2.1}$
	Single-Stage (EMP margin)	$39\%_{\pm 2.6}$	62.8%	$80\%_{\pm 1.9}$	$34\%_{\pm 2.9}$	57.4%	$79\%_{\pm 2.1}$	$43\%_{\pm 2.5}$	66.9%	$82\%_{\pm 1.7}$	$38\%_{\pm 2.7}$	61.3%	$80\%_{\pm 1.9}$
	Single-Stage (KL margin)	$43\%_{\pm 2.5}$	67.6%	$82\%_{\pm 1.7}$	$38\%_{\pm 2.8}$	62.8%	$80\%_{\pm 1.9}$	$47\%_{\pm 2.3}$	71.4%	$83\%_{\pm 1.6}$	$42\%_{\pm 2.6}$	66.1%	$82\%_{\pm 1.8}$
	Single-Stage (Vote)	$41\%_{\pm 2.6}$	64.7%	$81\%_{\pm 1.8}$	$36\%_{\pm 2.8}$	59.6%	$79\%_{\pm 2.0}$	$45\%_{\pm 2.4}$	68.3%	$82\%_{\pm 1.6}$	$40\%_{\pm 2.6}$	63.4%	$81\%_{\pm 1.8}$
	Two-Stage (EMP margin)	$49\%_{\pm 2.3}$	**91.4%**	$85\%_{\pm 1.4}$	$44\%_{\pm 2.5}$	**90.3%**	$84\%_{\pm 1.6}$	$53\%_{\pm 2.1}$	**92.8%**	$86\%_{\pm 1.2}$	$48\%_{\pm 2.4}$	**91.1%**	$85\%_{\pm 1.5}$
	Two-Stage (KL margin)	$55\%_{\pm 2.1}$	**95.2%**	$87\%_{\pm 1.2}$	$50\%_{\pm 2.3}$	**93.8%**	$86\%_{\pm 1.4}$	$59\%_{\pm 1.9}$	**95.7%**	$88\%_{\pm 1.0}$	$54\%_{\pm 2.1}$	**94.6%**	$87\%_{\pm 1.3}$
	Two-Stage (Vote)	$52\%_{\pm 2.2}$	**93.6%**	$86\%_{\pm 1.3}$	$47\%_{\pm 2.4}$	**92.1%**	$85\%_{\pm 1.5}$	$56\%_{\pm 2.0}$	**93.9%**	$87\%_{\pm 1.1}$	$51\%_{\pm 2.2}$	**92.7%**	$86\%_{\pm 1.4}$
Llama-3 70B	No Selection	100%	0%	$50\%_{\pm 1.1}$	100%	0%	$45\%_{\pm 1.3}$	100%	0%	$53\%_{\pm 1.0}$	100%	0%	$49\%_{\pm 1.2}$
	Single-Stage (EMP)	$39\%_{\pm 2.7}$	60.6%	$80\%_{\pm 1.9}$	$34\%_{\pm 3.0}$	55.3%	$78\%_{\pm 2.1}$	$43\%_{\pm 2.6}$	64.8%	$82\%_{\pm 1.7}$	$38\%_{\pm 2.8}$	59.7%	$80\%_{\pm 2.0}$
	Single-Stage (EMP margin)	$44\%_{\pm 2.5}$	65.4%	$81\%_{\pm 1.8}$	$39\%_{\pm 2.8}$	60.7%	$79\%_{\pm 2.0}$	$48\%_{\pm 2.4}$	69.6%	$83\%_{\pm 1.6}$	$43\%_{\pm 2.6}$	64.8%	$81\%_{\pm 1.9}$
	Single-Stage (KL margin)	$48\%_{\pm 2.4}$	70.3%	$83\%_{\pm 1.7}$	$43\%_{\pm 2.6}$	65.8%	$81\%_{\pm 1.9}$	$52\%_{\pm 2.2}$	74.6%	$84\%_{\pm 1.5}$	$47\%_{\pm 2.4}$	69.4%	$83\%_{\pm 1.7}$
	Single-Stage (Vote)	$46\%_{\pm 2.5}$	67.8%	$82\%_{\pm 1.7}$	$41\%_{\pm 2.7}$	62.4%	$80\%_{\pm 2.0}$	$50\%_{\pm 2.3}$	71.9%	$84\%_{\pm 1.6}$	$45\%_{\pm 2.5}$	66.3%	$82\%_{\pm 1.8}$
	Two-Stage (EMP margin)	$55\%_{\pm 2.1}$	**92.1%**	$86\%_{\pm 1.3}$	$50\%_{\pm 2.4}$	**90.6%**	$85\%_{\pm 1.5}$	$59\%_{\pm 2.0}$	**93.4%**	$87\%_{\pm 1.1}$	$54\%_{\pm 2.2}$	**91.8%**	$86\%_{\pm 1.4}$
	Two-Stage (KL margin)	$62\%_{\pm 1.9}$	**95.4%**	$88\%_{\pm 1.1}$	$57\%_{\pm 2.1}$	**94.1%**	$87\%_{\pm 1.3}$	$66\%_{\pm 1.8}$	**96.3%**	$89\%_{\pm 0.9}$	$61\%_{\pm 2.0}$	**95.2%**	$88\%_{\pm 1.2}$
	Two-Stage (Vote)	$59\%_{\pm 2.0}$	**93.8%**	$87\%_{\pm 1.2}$	$54\%_{\pm 2.2}$	**92.6%**	$86\%_{\pm 1.4}$	$63\%_{\pm 1.9}$	**94.9%**	$88\%_{\pm 1.0}$	$58\%_{\pm 2.1}$	**93.4%**	$87\%_{\pm 1.3}$

Table 6: **Two-stage vs. single-stage (additional judges)** (m=10, target agreement = 0.81, δ=0.10). Same format as Table 2. Coverage and Agr. are reported as mean ± std over 1,000 random splits.

Dataset	Cascade Configuration	Single-Stage		**Two-Stage (Ours)**				
		Cov.	GSR	Cov.	GSR	Comp. (%)		
						T_1	T_2	T_3
Chatbot Arena	Qwen2.5-72B only	$44\%_{\pm 2.6}$	67.4%	$58\%_{\pm 2.2}$	94.6%	–	–	100.0
	Qwen-7B → 32B → 72B	$55\%_{\pm 3.0}$	61.8%	$74\%_{\pm 2.1}$	**92.3%**	38.5	35.2	26.3
	Mistral-7B → DS-67B → GPT-120B	$53\%_{\pm 3.2}$	60.4%	$73\%_{\pm 2.3}$	**91.7%**	35.8	37.9	26.3
	DeepSeek-16B → Qwen-72B	$52\%_{\pm 3.4}$	59.6%	$70\%_{\pm 2.5}$	**90.8%**	46.1	53.9	–
HH-RLHF	Qwen2.5-72B only	$52\%_{\pm 2.3}$	75.3%	$66\%_{\pm 1.8}$	96.2%	–	–	100.0
	Qwen-7B → 32B → 72B	$64\%_{\pm 2.5}$	69.4%	$81\%_{\pm 1.8}$	**93.6%**	37.1	36.0	26.9
	Mistral-7B → DS-67B → GPT-120B	$62\%_{\pm 2.7}$	67.8%	$80\%_{\pm 2.0}$	**92.8%**	34.7	38.5	26.8
	DeepSeek-16B → Qwen-72B	$60\%_{\pm 2.9}$	66.3%	$77\%_{\pm 2.2}$	**91.6%**	44.8	55.2	–
AlpacaEval	Qwen2.5-72B only	$49\%_{\pm 2.5}$	71.2%	$63\%_{\pm 2.0}$	95.1%	–	–	100.0
	Qwen-7B → 32B → 72B	$58\%_{\pm 2.8}$	63.6%	$76\%_{\pm 2.0}$	**92.4%**	38.0	35.8	26.2
	Mistral-7B → DS-67B → GPT-120B	$56\%_{\pm 3.0}$	62.3%	$75\%_{\pm 2.2}$	**91.8%**	35.5	38.1	26.4
	DeepSeek-16B → Qwen-72B	$54\%_{\pm 3.1}$	61.7%	$72\%_{\pm 2.3}$	**90.9%**	45.9	54.1	–

Table 7: **Cascaded evaluation on additional datasets** (m=10, target agreement = 0.81, δ=0.10). Same format as Table 3.

I Full Per-Model and Per-Dataset Results

This appendix supplements the main experiments (Sections 4.2 and 4.3) with results for additional judge models not shown in the main tables. Table 2 reports three representative judges from distinct families (Llama-3-8B, Qwen2.5-72B, GPT-OSS-120B); here we present the same evaluations for three further judges—Mistral-7B, Qwen2.5-32B, and Llama-3-70B—covering the remaining model families and intermediate parameter scales. Results for the other three judges (Qwen2.5-7B, DeepSeek-16B, DeepSeek-67B) follow the same trends and are omitted for brevity.

Ranking loss. Table 5 extends the monotonicity analysis of Table 1 to the additional judges. The same pattern holds: single-stage ranking loss grows substantially with m, while both Stage I and Stage II maintain near-zero ranking loss across all settings.

Two-stage vs. single-stage. Table 6 extends the main comparison of Table 2 to the additional judges. All trends are consistent: every two-stage variant exceeds the $1-\delta = 90\%$ GSR target, while all single-stage baselines fall short. Two-stage coverage improvements over the matched single-stage baseline range from 8–14 percentage points across all judges and datasets. Beyond the tabular results, Figures 8 to 10 extend the guarantee-curve and

(Judge, Dataset, m)	λ=0.0	λ=0.2	λ=0.4	λ=0.6	λ=0.8
Llama-3-8B, TL;DR, m=5	9.6%	5.8%	3.1%	1.5%	0.6%
Llama-3-8B, Chatbot Arena, m=10	9.9%	6.5%	3.7%	1.7%	0.7%
Qwen2.5-72B, TL;DR, m=10	8.4%	5.2%	2.9%	1.3%	0.5%
GPT-OSS-120B, TL;DR, m=10	7.1%	4.3%	2.2%	0.9%	0.3%
GPT-OSS-120B, Chatbot Arena, m=10	8.2%	5.0%	2.7%	1.1%	0.4%
GPT-OSS-120B, HH-RLHF, m=10	6.5%	3.9%	1.9%	0.8%	0.3%

Table 8: **Direct validation of the cross-stage monotonicity condition.** Conditional shortlist miss probability (%) as a function of the Stage II margin threshold.

Method	GSR	Cov.
Jung et al. (m>2 adapted) + variance normalization	74.2%	53%
Single-Stage (KL margin)	76.4%	56%
Two-Stage (KL margin)	**96.8%**	**70%**

Table 9: **Strengthened single-stage comparison** on TL;DR with GPT-OSS-120B (m=10, target agreement = 0.81), averaged over 1,000 splits.

m-effect analysis of Figure 4(a)–(b) to additional (dataset, model) combinations. The same qualitative conclusions hold: the two-stage empirical agreement consistently tracks above the diagonal across the full 70–95% target range, and two-stage GSR remains above the $1-\delta = 90\%$ threshold as m grows from 5 to 20, whereas single-stage GSR degrades sharply.

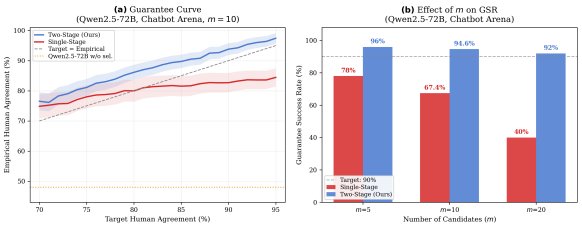

Figure 8: **Guarantee curve and m-effect on Chatbot Arena** (Qwen2.5-72B). Same layout as Figure 4(a)–(b).

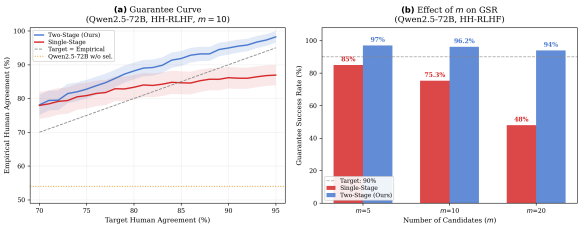

Figure 9: **Guarantee curve and m-effect on HH-RLHF** (Qwen2.5-72B). Same layout as Figure 4(a)–(b).

Cascaded architecture on additional datasets. Table 3 in the main text reports cascade results on TL;DR. Table 7 extends the evaluation to the remaining three datasets (Chatbot Arena, HH-RLHF, AlpacaEval) using representative cascade configurations. The conclusions are fully consistent: all two-stage cascades exceed the $1-\delta = 90\%$ GSR threshold, while all single-stage cascades fall short. Cross-dataset variation is modest; three-tier two-stage cascades cover 73–81% of instances and achieve 92–94% GSR uniformly. The tier-composition percentages are also stable across datasets: combining Table 3 and Table 7, the weakest tier handles 32–39% and the strongest tier handles 26–29% of covered instances in three-tier configurations, confirming that the computational savings observed on TL;DR generalize broadly. Figure 11 further visualizes the cascaded guarantee curve and evaluator composition on Chatbot Arena, showing that the empirical agreement tracks the target throughout the 70–95% range and that weaker tiers handle the majority of instances at lenient targets.

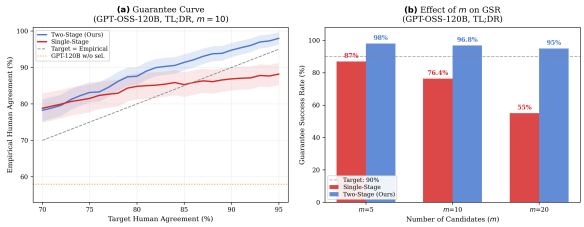

Figure 10: **Guarantee curve and m-effect on TL;DR** (GPT-OSS-120B). Same layout as Figure 4(a)–(b).

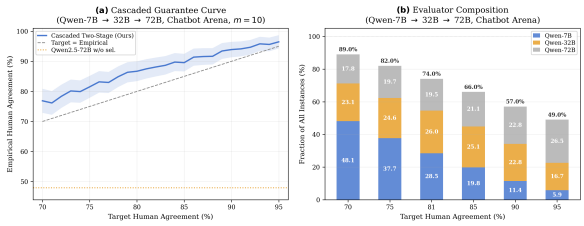

Figure 11: **Cascaded evaluation on Chatbot Arena** (Qwen-7B → 32B → 72B, m=10). Same layout as Figure 4(c)–(d).

Variant	Renorm.	Exp.	Cond. Avg.	Var. Norm.	Scale	GSR	Cov.
Full KL margin	✓	✓	✓	✓	✓	**96.8%**	**70%**
− Renormalization	–	✓	✓	✓	✓	83.5%	57%
− Variance normalization	✓	✓	✓	–	✓	89.2%	61%
− Scale factor	✓	✓	✓	✓	–	91.4%	64%
− Conditional averaging	✓	✓	–	✓	✓	92.6%	65%
− Exponential transform	✓	–	✓	✓	✓	95.1%	68%

Table 10: **Component-wise KL-margin ablation** under the two-stage framework on TL;DR with GPT-OSS-120B (m=10, target agreement $= 0.81$), averaged over 1,000 splits.