CAN WE TRUST THIS EVALUATION? INSTANCE-LEVEL RELIABILITY OF LLM-AS-A-JUDGE

Anonymous authors
Paper under double-blind review

ABSTRACT

LLM-as-a-judge evaluation commonly aggregates multiple judges' decisions into a majority vote, yet this panel decision process does not guarantee that the evaluation of an individual item is reliable. We propose Structured Cross-fitted Uncertainty Risk (SCUR), a label-free approach for estimating instance-level evaluation reliability based on the judge panel's behavior. A latent measurement model serves as the backbone for measurement system analysis, from which SCUR derives three complementary signals capturing panel uncertainty, measurement cross-entropy, and perturbation sensitivity. These three sources of evidence are combined into a risk score. We evaluate the SCUR approach on five real-world benchmarks using majority-vote error ranking, selective trust, and risk stratification. Furthermore, we show our approach to be more effective at detecting high-consensus risk items, which are the blind spots for many other LLM-as-a-judge evaluation methods.

1 INTRODUCTION

LLM-as-a-judge evaluation has become a standard approach for assessing open-ended generation tasks, including dialogue, reasoning, summarization, coding, instruction following, and safety (Liu et al., 2023; Lin & Chen, 2023; Li et al., 2025), because it is scalable, inexpensive, and easy to deploy. However, since the judge is itself a large language model (LLM), it can be biased, unstable, prompt-sensitive, and overconfident. A natural solution is to use multiple LLM judges to form a judge panel (Zheng et al., 2023; Verga et al., 2024; Chan et al., 2024) and output only the majority-vote decision. A typical pipeline asks multiple LLMs to select one answer from two options—option A or B—for a single question or prompt, then aggregates their votes into a label or score. This raises a fundamental question: *How can we trust this panel's decision?*

Prior work has documented the option-display-position bias in LLM evaluators (i.e., the evaluator prefers a certain position, such as the first position, so leaning to A, instead of the real contents in these options (Wang et al., 2024a; Shi et al., 2025)), self- and model-family preference (i.e., different LLMs from the same foundation model family tend to vote similarly (Wataoka et al., 2024; Li et al., 2026)), and broader instability in LLM judgments (Schroeder & Wood-Doughty, 2024; Li et al., 2025). These concerns cannot be fully answered by simply aggregating multiple LLM judges' decisions (Zhang et al., 2023; Chan et al., 2024). Note that these LLM evaluators may share overlapping training corpora, similar RLHF preferences, internet writing norms, safety alignment policies, benchmark contamination, and stylistic priors. As a result, *multiple LLM judges may agree for the same wrong reason*. Thus, implicitly treating panel agreement as the evidence of answer correctness is very problematic (Wang et al., 2024a; Shi et al., 2025; Li et al., 2026; Schroeder & Wood-Doughty, 2024). It may create a failure mode which we denote as *"correlation collapse"*: a panel appears to be highly confident because most LLM judges agree with each other, but this agreement is driven by the shared judge bias rather than the underlying answer quality. In such cases, high consensus may produce false confidence, instead of a trustworthy decision. The existing LLM-evaluation aggregation methods focus on producing better decisions by weighting on each judge's reliability. In contrast, we study a complementary problem: given a panel's decision on one particular item, can we estimate how much this decision can be trusted?

Estimating the reliability of an evaluation panel's decision requires more than measuring the degree of panel agreement alone. We introduce a latent measurement model that employs cross-fitted,

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

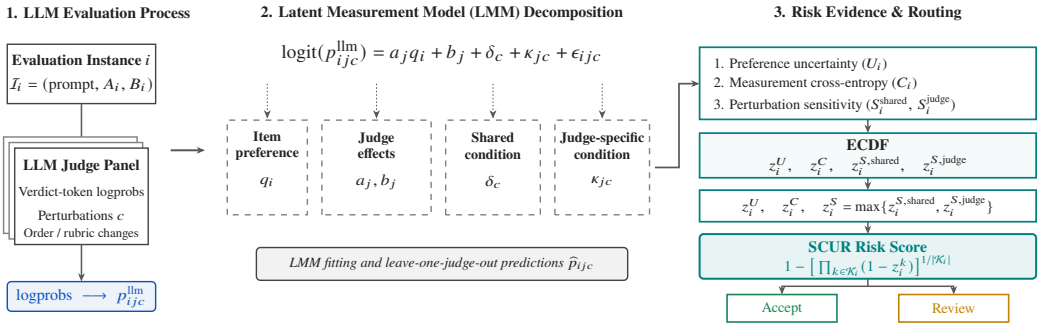

Figure 1: Overview of the proposed SCUR framework. The latent measurement model (LMM) decomposes LLM verdict probabilities into item, judge, and perturbation effects. Cross-fitted predictions yield preference uncertainty (U), measurement cross-entropy (C), and shared and judge-specific perturbation sensitivities. Each raw signal is separately normalized using its unlabeled reference ECDF; the two sensitivity percentiles are then combined by their maximum to obtain z^S. The available evidence z^U, z^C, z^S is fused into SCUR for automatic acceptance or human review.

judge-level predictive probabilities under original and perturbation conditions. From these predictions, the Structured Cross-fitted Uncertainty Risk (SCUR) score is formulated, which preserves three interpretable misjudgment mechanisms—the latent uncertainty among judges, the alignment of predictive vs. native preference probabilities, and the panel's sensitivity to perturbation conditions—and combines their reference percentiles into a label-free risk score via a complement-based geometric mean.

We evaluate SCUR on five real-world LLM-as-a-judge benchmarks spanning pairwise preference, objective correctness, and instruction following. These experiments examine whether the proposed risk score correctly identifies unreliable panel decisions, improves evaluation reliability through selective review, and produces meaningful risk stratification across evaluated instances.

Our primary contributions are as follows:

- We propose SCUR, a label-free approach to estimating decision reliability from a LLM-as-a-judge panel, without requiring ground-truth labels. It treats uncertainty, misaligned responses, and condition sensitivity as potential failure alarms for constructing a sensible risk score.
- We introduce an instance-level diagnostic analysis that decomposes the estimated risk into interpretable behavioral factors, providing insights into why a panel decision is deemed unreliable, then subsequently recommending human intervention.
- We conduct extensive experiments on five real-world LLM-as-a-judge benchmarks, covering pairwise preference, objective correctness, and instruction following, and demonstrate that SCUR consistently outperforms existing uncertainty-only baselines in terms of majority-vote error ranking, selective trust, and risk stratification. Furthermore, we demonstrate that SCUR is effective for identifying high-consensus risky items and transferable to other vote aggregation rules.

2 Related Work

LLM-as-a-judge evaluation. LLM-as-a-judge methods use strong language models to evaluate human- or machine-generated content through pairwise comparison, scalar scoring, or rubric-guided critique (Lin & Chen, 2023; Wang et al., 2024b; Kim et al., 2024). These methods are attractive because they scale beyond manual annotation, but they are not neutral measurement devices. Prior work documents several deficiencies including position bias, verbosity bias, self-enhancement, prompt sensitivity, and other evaluator artifacts (Zheng et al., 2023; Wang et al., 2024a; Shi et al., 2025). Recent work also studies multi-judge or multi-role evaluator networks as a way to improve alignment with human judgments (Zhang et al., 2023). Our work is closest in spirit to these reliability concerns, but shifts the target from average judge quality to the reliability of each panel decision on individual instances.

Noisy annotators and label aggregation. The proposed framework is related to a long line of noisy-label aggregation and crowdsourcing models. Dawid–Skene estimates latent labels together with annotator confusion matrices from redundant noisy annotations (Dawid & Skene, 1979). Subsequent methods, including GLAD (Whitehill et al., 2009), MACE (Hovy et al., 2013), and Bayesian annotation models (Paun et al., 2018), jointly model annotator competence, item difficulty, and annotation uncertainty to improve consensus estimation. These methods are designed to recover a more accurate latent label or aggregate decision. Recent work has improved LLM evaluation through confounder-aware panel aggregation (Zhao et al., 2026), item-response-theoretic modeling of judge behavior (Choi et al., 2026), and selective confidence estimation for pairwise judging (Badshah et al., 2026b). In contrast, we assume that a panel decision has already been produced and address a complementary problem: estimating the reliability of that individual panel decision without requiring ground-truth labels.

Pairwise preference modeling. Pairwise evaluation is related to Bradley-Terry, Luce-choice, and related ranking models (Bradley & Terry, 1952; Luce, 1959). These models infer latent preference or quality scores from comparisons. In our setting, the comparison is contaminated by the evaluator bias and random noise due to the stochastic nature of generative LLMs, so the observed vote cannot be treated as deterministic. We model the LLM judge process itself and use the fitted structure to quantify the trustworthiness of the resulting panel decision.

Measurement system reliability. This work is also related to measurement system analysis, especially gauge repeatability and reproducibility, which partition the total variation in measurements into variance components due to items, the measurement system, and repeated measurements (Automotive Industry Action Group, 2010). A related line of work studies annotation quality in crowdsourced data through consistency, annotator credibility, variance decomposition, and spam diagnostics (Ba et al., 2025). We adapt this perspective to LLM evaluation by treating items, judges, and perturbations as sources of measurement variation.

3 METHODOLOGY

We first introduce a latent measurement model (LMM) to model LLM-judge decisions under various designed perturbation schemes. We then extract three failure mechanism-specific risk signals for the judge panel—decision uncertainty, measurement cross-entropy, and perturbation sensitivity—and combine their reference percentiles into a risk-ranking score.

3.1 LATENT MEASUREMENT MODEL

Suppose there are two options for an LLM judge to choose from for each piece of item to be evaluated. Each item consists of a prompt, two candidate answers, and an optional judging rubric, and it is denoted by $\mathcal{I}_i = (\mathrm{prompt}_i, A_i, B_i, \mathrm{rubric}_i)$. For item i, judge j, and perturbation condition $c \in \mathcal{C}$, let $p_{ijc}^{\mathrm{llm}} \in (0, 1)$ be the judge's response probability that the *canonical* candidate A_i wins. p_{ijc}^{llm} is obtained by exponentiating and normalizing the judge's verdict-token logprobs, the logarithms of the probabilities the LLM assigns to the possible verdicts. We reserve $c = 0$ for the original condition: the unperturbed answer order, prompt, and rubric. When $c \neq 0$ it denotes one prespecified intervention, such as swapping the display order of answers or changing the judging rubric. After an answer-order swap, for example, the probability is mapped back to the same canonical event "original A_i wins," so probabilities remain comparable between the original and perturbed conditions.

p_{ijc}^{llm} has been widely used to measure how confident a large language model is in the verdict token it generates. We employ a latent measurement model (LMM) to decompose its log-odds, making item, judge, and condition effects separable:

$$\mathrm{logit}\big(p_{ijc}^{\mathrm{llm}}\big) = a_j q_i + b_j + \delta_c + \kappa_{jc} + \epsilon_{ijc}, \qquad \epsilon_{ijc} \sim \mathcal{N}(0, \sigma_j^2). \tag{1}$$

This decomposition distinguishes the item-level preference q_i from systematic judge and condition effects and residual variation, allowing individual judge responses to be assessed relative to their model-implied preferences. The latent variable q_i represents the panel's underlying consensus preference for the canonical candidate, inferred jointly from all judges' responses rather than a correctness oracle. The discrimination $a_j = \exp(\alpha_j) > 0$ controls how strongly judge j responds to

that consensus, as in a two-parameter item-response model (Lord, 1980); b_j is this judge's overall tendency toward canonical candidate A. The shared perturbation effect δ_c captures the panel-wide response to intervention c, whereas κ_{jc} captures judge j's deviation from that shared response. We set $\delta_0 = \kappa_{j0} = 0$, so the unperturbed original item serves as a reference.

Estimation. We estimate the LMM mean structure using alternating penalized least-squares updates with equal observation weights. The quadratic penalties admit a Gaussian-shrinkage interpretation, shrinking $q_i, b_j, \delta_c, \kappa_{jc}$ toward zero and a_j toward one. In particular, shrinking κ_{jc} toward zero regularizes each judge's condition effect toward the shared effect δ_c. Judge-specific residual variances are estimated after fitting the mean structure. Appendix A provides the penalty strengths, scale and centering conventions, and optimization details.

Dataset partition and cross-fitting. We partition each benchmark by items in a 50/30/20 ratio into $\mathcal{D}_{\text{fit}}/\mathcal{D}_{\text{ref}}/\mathcal{D}_{\text{eval}}$ datasets: $\mathcal{D}_{\text{fit}}$ estimates LMM parameters, $\mathcal{D}_{\text{ref}}$ supplies the reference distributions for risk normalization, and $\mathcal{D}_{\text{eval}}$ is held out for evaluation. No panel decision-error labels are used for LMM fitting or risk estimation. For each reference or evaluation item, we freeze the fitted judge and condition parameters and infer $q_i^{(-j)}$ from other judges, excluding judge j's responses for that item. The cross-fitted prediction is $\widehat{p}_{ijc}^{(-j)} = \sigma(a_j q_i^{(-j)} + b_j + \delta_c + \kappa_{jc})$, where σ is the logistic function; for simplicity, we rewrite it as $\widehat{p}_{ijc}$. Excluding judge j's own responses ensures that this prediction is based on evidence from the remaining panel, preventing a prediction contamination due to self-response reinforcement.

3.2 Evidence-Based Risk Signals

We define three complementary risk signals: U measures ambiguity in the pooled preference, C assesses individual judge responses against the learned panel preference structure, and S measures dependence on perturbation conditions.

Model-based preference uncertainty. Let $\overline{p}_i = J_i^{-1} \sum_{j=1}^{J_i} \widehat{p}_{ij0}$, where J_i is the number of available judges for item i, be the mean cross-fitted preference probability under the original condition. We define

$$U_i = H(\overline{p}_i), \qquad H(p) = -p \log p - (1-p) \log(1-p).$$

U_i is largest when $\overline{p}_i$ is close to 0.5, meaning that the panel does not clearly favor either candidate. This may occur because individual judges are uncertain or because different judges confidently favor different candidates. However, it may remain low when the judges confidently agree and therefore cannot identify errors shared across the panel.

Measurement cross-entropy. We compare each judge's LLM confidence p_{ij0}^{llm} with its cross-fitted LMM prediction $\widehat{p}_{ij0}$ under the original condition using binary cross-entropy:

$$C_i = -\frac{1}{J_i} \sum_{j=1}^{J_i} \left[p_{ij0}^{\text{llm}} \log \widehat{p}_{ij0} + (1 - p_{ij0}^{\text{llm}}) \log(1 - \widehat{p}_{ij0}) \right].$$

C_i increases when a judge's LLM confidence disagrees with the cross-fitted prediction based on the remaining panel. The mismatch is greater when the judge confidently favors one candidate while the cross-fitted prediction favors the other. Such a mismatch indicates that the judge's confidence is not well supported by the preference structure inferred from the remaining panel, making it informative for assessing decision risk. Cross-entropy also reflects the uncertainty of the LLM confidence itself, so C_i can be high even when the two probabilities are similar but uncertain. Thus, C_i reflects both individual-judge uncertainty and cross-fitted mismatch, complementing the pooled preference uncertainty measured by U_i.

Perturbation sensitivity. For benchmarks with paired perturbations, where the same item is evaluated under the original and perturbed conditions, let $\mathcal{C}_i^+ = \mathcal{C}_i \setminus \{0\}$ denote the available perturbed conditions, such as answer-order swaps, strict-rubric changes, and evidence-first rubric changes.

Let $\eta_{ij0} = a_j q_i^{(-j)} + b_j$. Keeping the cross-fitted item margin fixed, we separately measure the probability changes due to the shared and judge-specific condition effects:

$$S_i^{\text{shared}} = \frac{1}{J_i|\mathcal{C}_i^+|} \sum_{j=1}^{J_i} \sum_{c \in \mathcal{C}_i^+} \left|\sigma(\eta_{ij0} + \delta_c) - \sigma(\eta_{ij0})\right|,$$

$$S_i^{\text{judge}} = \frac{1}{J_i|\mathcal{C}_i^+|} \sum_{j=1}^{J_i} \sum_{c \in \mathcal{C}_i^+} \left|\sigma(\eta_{ij0} + \delta_c + \kappa_{jc}) - \sigma(\eta_{ij0} + \delta_c)\right|.$$

S_i^{shared} measures the probability change associated with a perturbation across the panel, while S_i^{judge} measures additional judge-specific changes beyond this shared effect. These two components capture complementary forms of perturbation sensitivity and are combined after reference normalization, as described below. Large changes indicate that a preference depends on the judging conditions, which can undermine its stability.

3.3 Label-Free Risk Score

Reference normalization. We use r_i^k for raw signal k and z_i^k for its normalized evidence. Specifically, let $r_i^U = U_i$, $r_i^C = C_i$, $r_i^{S,\text{shared}} = S_i^{\text{shared}}$, and $r_i^{S,\text{judge}} = S_i^{\text{judge}}$. For each available raw signal k, we estimate an empirical cumulative distribution function (ECDF) (Wasserman, 2004) on the unlabeled reference split:

$$z_i^k = \widehat{F}_{k,\mathcal{D}_{\text{ref}}}(r_i^k) = \frac{1}{|\mathcal{D}_{\text{ref}}^k|} \sum_{h \in \mathcal{D}_{\text{ref}}^k} \mathbf{1}\{r_h^k \le r_i^k\},$$

where $\mathcal{D}_{\text{ref}}^k$ contains reference items with risk signal k. The ECDF is estimated without golden (true) labels, and it is then frozen for evaluation. The transformation from r_i^k to z_i^k maps heterogeneous raw signal scales to comparable percentiles. For perturbation sensitivity, we then combine its two components as

$$z_i^S = \max\left\{z_i^{S,\text{shared}}, z_i^{S,\text{judge}}\right\}.$$

Taking the maximum after normalization retains whichever form of perturbation sensitivity is more unusual relative to its reference distribution. Therefore, the normalized evidence captures three complementary aspects of decision risk: preference ambiguity (z_i^U), individual-judge uncertainty and mismatch (z_i^C), and perturbation sensitivity (z_i^S).

Structured Cross-fitted Uncertainty Risk. Let $\mathcal{K}_i$ contain the available channels: $\{U, C, S\}$ with paired perturbations and $\{U, C\}$ otherwise. We construct *Structured Cross-fitted Uncertainty Risk* (SCUR) as the complement of the geometric mean of the complements of normalized evidence:

$$\text{SCUR}_i = 1 - \left[\prod_{k \in \mathcal{K}_i} (1 - z_i^k)\right]^{1/|\mathcal{K}_i|}. \tag{2}$$

Specifically, $\text{SCUR}_i = 1 - [(1 - z_i^U)(1 - z_i^C)(1 - z_i^S)]^{1/3}$. The complement-based geometric mean aggregates the available risk evidence: a high percentile for any evidence type reduces its complement and increases the resulting SCUR score. The $|\mathcal{K}_i|$-th root accounts for the number of available evidence types, allowing the same aggregation rule when perturbation sensitivity is unavailable. The resulting $\text{SCUR}_i \in [0, 1]$ provides a relative risk score, which will be used to rank items for automatic acceptance or human review.

4 Experiments

4.1 Benchmark Evaluation

Benchmarks. We validate the proposed reliability framework on five multi-LLM-as-a-judge benchmarks: Chatbot-Arena (Zheng et al., 2023), HaluEval (Li et al., 2023),

Table 1: **Majority-vote error ranking** (Mean ± standard deviation over five independent trials). Panel Err. is the original-condition majority-vote error rate. Err@50 is the error rate among the lowest-risk 50% of evaluation items. SCUR outperforms baselines on each benchmark.

Benchmark	Sample Sizes	#Judges	Panel Err.	Method	AUROC ↑	$1 - \mathrm{AURC}$ ↑	Err@50 ↓
Chatbot-Arena	#Obs.: 589,050 #Items: 17,325 #Ref.: 5,198 #Test: 3,465	17	.3427 ± .0082	PE	.674 ± .008	.773 ± .010	.229 ± .014
				VE	.625 ± .007	.741 ± .008	.252 ± .011
				DS entropy	.673 ± .006	.765 ± .009	.222 ± .010
				MSP	.647 ± .011	.762 ± .012	.241 ± .011
				SCUR	.696 ± .007	.796 ± .008	.195 ± .008
HaluEval	#Obs.: 120,000 #Items: 20,000 #Ref.: 6,000 #Test: 4,000	6	.1937 ± .0035	PE	.728 ± .006	.891 ± .004	.081 ± .005
				VE	.667 ± .003	.868 ± .004	.129 ± .003
				DS entropy	.674 ± .004	.868 ± .004	.122 ± .003
				MSP	.718 ± .011	.897 ± .005	.093 ± .006
				SCUR	.732 ± .006	.901 ± .005	.079 ± .005
MMLU	#Obs.: 477,428 #Items: 28,084 #Ref.: 8,425 #Test: 5,617	17	.2295 ± .0046	PE	.787 ± .004	.907 ± .003	.079 ± .003
				VE	.744 ± .005	.892 ± .003	.098 ± .005
				DS entropy	.730 ± .007	.886 ± .003	.093 ± .003
				MSP	.693 ± .006	.878 ± .003	.129 ± .005
				SCUR	.794 ± .005	.915 ± .003	.062 ± .003
LLMBar	#Obs.: 12,570 #Items: 419 #Ref.: 126 #Test: 83	5	.3711 ± .0198	PE	.623 ± .048	.711 ± .029	.290 ± .031
				VE	.585 ± .046	.677 ± .028	.311 ± .034
				DS entropy	.640 ± .030	.704 ± .019	.265 ± .013
				MSP	.498 ± .050	.651 ± .027	.381 ± .029
				SCUR	.654 ± .042	.732 ± .021	.238 ± .017
LLMEval^2	#Obs.: 6,000 #Items: 200 #Ref.: 60 #Test: 40	5	.2100 ± .0418	PE	.553 ± .068	.810 ± .056	.190 ± .055
				VE	.523 ± .086	.797 ± .055	.197 ± .040
				DS entropy	.546 ± .064	.807 ± .034	.200 ± .041
				MSP	.603 ± .122	.824 ± .077	.170 ± .076
				SCUR	.632 ± .070	.861 ± .036	.160 ± .042

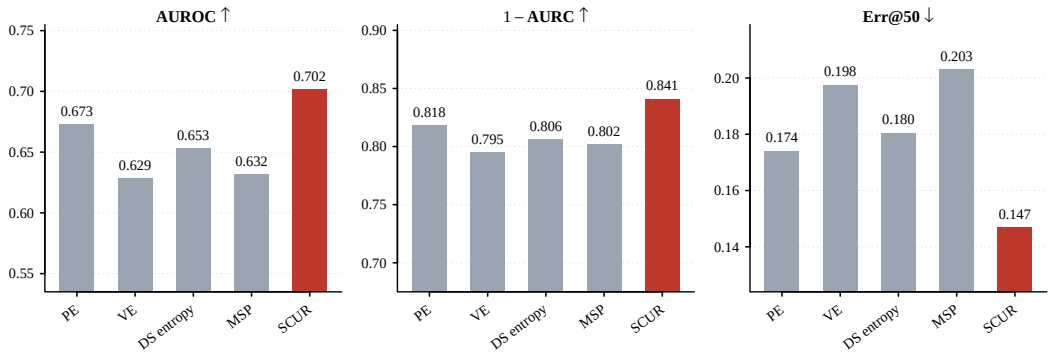

Figure 2: **Average performance across five benchmarks.** Each benchmark receives equal weight. SCUR (red) has the highest AUROC and $1 - \mathrm{AURC}$ and the lowest Err@50; gray bars show the four baselines.

MMLU (Hendrycks et al., 2021), LLMBar (Zeng et al., 2024; Gao et al., 2024), and LLMEval^2 (Zhang et al., 2023; Gao et al., 2024; Zeng et al., 2024). They cover pairwise preference, hallucination detection, multiple-choice correctness, instruction following, and multidimensional prompt-following. We use a 50/30/20 data split: 50% fits the LMM, 30% supplies the reference distributions for SCUR, and 20% is untouched test data. We run 5 independent trials.

Chatbot-Arena, HaluEval, and MMLU use existing judge outputs. Chatbot-Arena includes original and swapped answer orders; HaluEval and MMLU contain only the original condition. For LLMBar and LLMEval^2, we collect verdict-token log-probabilities from five judge models, crossing original/swapped answer order with standard, strict, and evidence-first rubrics to obtain six conditions per item–judge pair. Thus Chatbot-Arena, LLMBar, and LLMEval^2 provide U, C, S, whereas HaluEval and MMLU provide only U, C. See Appendix C.2 for the benchmark-specific conditions and judging protocol.

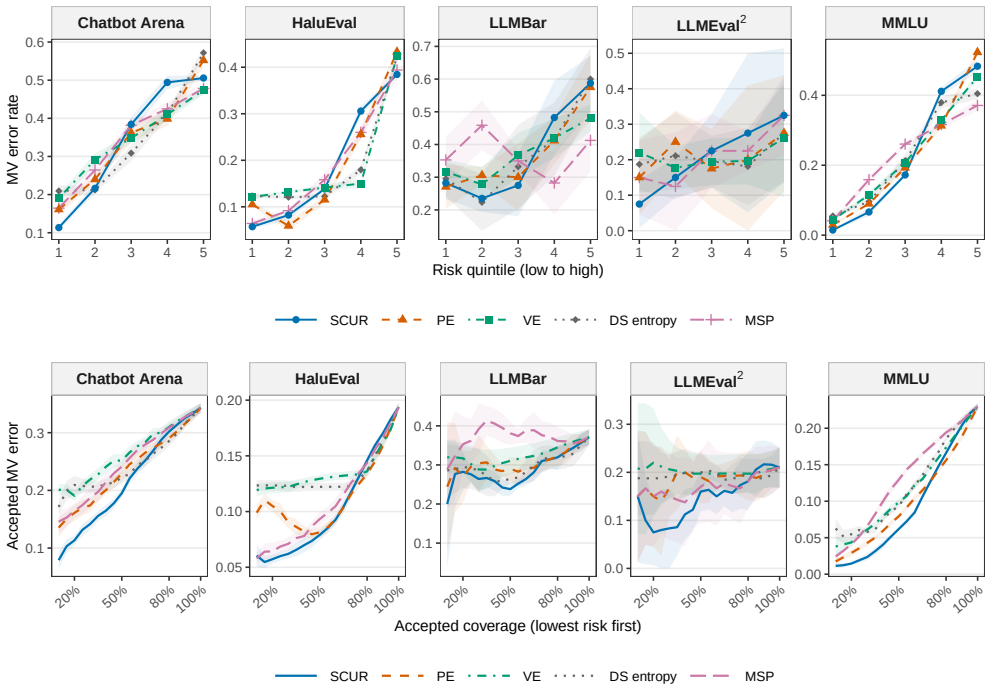

Figure 3: **Majority-vote risk stratification and selective trust.** Top: error across increasing risk quintiles; monotonic increases indicate that higher-risk quintiles contain more errors. Bottom: error among the lowest-risk items retained at each coverage; lower curves indicate fewer errors at the same coverage. Lines show five-seed means, with shaded bands indicating ± 1 standard deviation.

Risk baselines. We compare SCUR with four panel-level uncertainty baselines: vote entropy (VE), which is the binary entropy of the majority-vote rate related to vote-agreement confidence used for LLM judges (Radharapu et al., 2026); predictive entropy (PE), which is the entropy of the panel-mean probability, extending predictive entropy used for LLM-judge abstention (Badshah et al., 2026a); DS entropy, which is the posterior entropy of the Dawid-Skene model, adapting its LLM-evaluator aggregation applications (Gao et al., 2024; Zhao et al., 2026) to risk scoring; and mean per-judge MSP uncertainty (Hendrycks & Gimpel, 2017; Radharapu et al., 2026). DS entropy ranks items by uncertainty; DS aggregation, discussed later in this section, instead determines their verdicts. Appendix B specifies the formulas and the panel adaptations of these prior studies.

Evaluation metrics. Our primary target is the error made by majority voting verdict, $e_i^{\mathrm{MV}} = \mathbf{1}\{\widehat{y}_i^{\mathrm{MV}} \neq y_i^{\mathrm{gold}}\}$. AUROC measures how well risk separates incorrect from correct verdicts; we report $1-\mathrm{AURC}$, the complement of area under the risk–coverage curve (Geifman et al., 2019; Zhou et al., 2025), equivalently the mean retained accuracy over all coverage levels k/n, $k = 1, \ldots, n$ (higher is better); and Err@50 is the error rate among the lowest-risk half of evaluation items. We also examine error by increasing risk quintile and across retained coverage. For an alternative aggregation rule g (Section 4.3), we recompute the error as $e_i^g = \mathbf{1}\{\widehat{y}_i^g \neq y_i^{\mathrm{gold}}\}$, while using the same SCUR scores to rank the items.

Majority-vote error ranking. Table 1 shows that SCUR has the highest mean AUROC and $1 - \mathrm{AURC}$ and the lowest mean Err@50 on each benchmark. Figure 2 summarizes the equally weighted benchmark averages. Compared with PE, the best baseline on average, AUROC increases from .673 to .702, $1 - \mathrm{AURC}$ from .818 to .841, and Err@50 decreases from .174 to .147. The improvement in AUROC shows that SCUR better separates incorrect from correct panel decisions. The lower Err@50 gives this ranking gain an operational meaning: at the same 50% acceptance budget, the retained set contains fewer errors. The $1 - \mathrm{AURC}$ improvement shows that this advantage persists

across coverage levels rather than at a single cutoff. Overall, SCUR provides a more effective ranking of panel-error risk than the uncertainty baselines.

Selective trust and risk strata. Figure 3 examines how observed error changes with SCUR risk and retained coverage. On Chatbot-Arena, HaluEval, and MMLU, error increases monotonically from the lowest to the highest SCUR quintile, showing that higher SCUR scores correspond to higher empirical error. LLMEval^2 shows the same monotonic pattern, while LLMBar shows some variation among the lower-risk quintiles but substantially higher error in the two highest-risk groups.

Across the coverage curves, SCUR generally achieves lower error at the same coverage than the baselines, particularly at lower coverage. On LLMBar and LLMEval^2, which have substantially fewer evaluation items than the other benchmarks, estimates are more variable and the baselines show weak or inconsistent ordering across risk quintiles. Despite this increased variability, SCUR retains a clearer separation between lower- and higher-risk items.

4.2 High-Consensus Blind Spots

High-consensus (HC) errors are challenging cases for any disagreement-based risk measurement, because incorrect verdicts can remain hidden when most judges agree on this incorrect choice. We therefore evaluate items for which at least 80% of judges choose the same answer under the original condition, using each judge's binary A/B choice. This defines the high-consensus (HC) subset; gold labels are used only to determine whether the majority verdict is correct. We ask whether SCUR can still distinguish incorrect from correct verdicts by marking them with high risk.

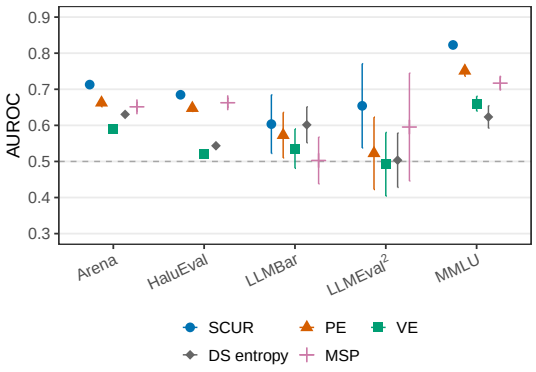

Figure 4: **Performance on high-consensus items.** AUROC on items with at least 80% vote agreement. Points show five-seed means and whiskers indicate ± 1 standard deviation. The dashed line marks chance-level AUROC.

Figure 4 shows that SCUR achieves the highest mean AUROC on the HC subset across all five benchmarks. The advantage is particularly clear on the larger benchmarks, while estimates are more variable on LLMBar and LLMEval^2. These results show that SCUR can identify risk differences even among high-consensus decisions, where other methods struggle to separate incorrect verdicts from correct verdicts.

The matched-vote analysis in Appendix D provides a stricter test by comparing items with identical original-condition vote patterns across judges (e.g., a 4-to-1 vote is compared only with other 4-to-1 votes in which the same four judges support the same candidate). The appendix also considers an alternative definition of high consensus based on panel-mean probabilities, selecting items with a panel-mean confidence of at least .8. Across both settings, SCUR outperforms the baselines in separating incorrect from correct verdicts, indicating that SCUR consistently identifies panel errors that remain difficult to detect under high consensus.

4.3 Risk Generalizability

SCUR assesses the reliability of panel decisions independently of how judge evaluations are aggregated, allowing it to be applied to different aggregation rules. We evaluate this setting with Dawid–Skene (DS), MACE, GLAD, and CARE (Zhao et al., 2026) in addition to majority vote. Each method produces its own verdict and error labels.

Figure 5 shows that retaining lower-risk decisions according to SCUR improves accuracy across all five benchmarks and five aggregation rules, with an average gain of 4.6 pp at 80% lowest-risk threshold and 12.2 pp at 50% threshold, compared with accepting all panel decisions. These results show that SCUR is robust to different aggregation methods; therefore, it can serve as a common risk-assessment metric for either accepting the panel decision or moving the case to human review.

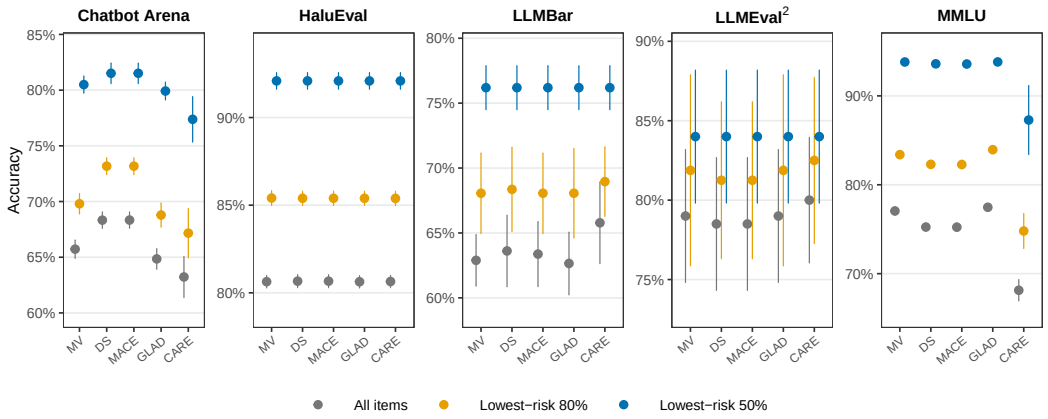

Figure 5: **Selective acceptance across five aggregation rules.** Gray: all evaluation items; orange/blue: lowest-risk 80%/50% based on SCUR ranking. Points and whiskers show five-seed means and ± 1 standard deviation; y-axis ranges vary by benchmark.

Table 2: **Instance-level diagnostics on held-out items.** Panel/Gold reports the majority verdict and gold label. Vote agreement is the fraction of judges supporting the original-condition majority verdict. The z values are reference percentiles, and – indicates that the corresponding risk evidence is unavailable.

Benchmark	Instance ID	Panel/Gold	Vote agreement	z^U	z^C	z^S	SCUR	Interpretation / action
Chatbot Arena	1110836523	A/B	100%	.652	.482	1.000	.967	Extreme perturbation sensitivity despite unanimity. **Review:** unstable across conditions.
HaluEval	6411	B/A	83.3%	.999	.988	–	.996	High preference ambiguity and cross-entropy. **Review:** strong risk evidence.
LLMBar	61	B/A	60%	.841	.778	.754	.794	Multiple moderately elevated signals. **Review:** combined risk evidence.
LLMEval^2	22	A/B	80%	.983	.967	.833	.955	All three risk signals are elevated. **Review:** high risk despite agreement.
MMLU	19396	A/A	94.1%	.001	.000	–	.001	Low preference ambiguity and cross-entropy. **Accept:** low review priority.

4.4 Panel-Decision Diagnosis

Table 2 illustrates how the three risk signals contribute to individual review decisions. High z^U, z^C, and z^S indicate preference ambiguity, judge-level mismatch, and sensitivity to perturbations, respectively; high z^C may also reflect uncertain LLM confidence. Together, the signals provide complementary evidence for why an item may warrant review without requiring a gold label.

The examples listed in the table show that elevated risk can arise under strong panel agreement, without perturbation evidence, or through several moderately elevated signals rather than a single dominant one. The last row (MMLU, Instance ID 19396) provides a low-risk contrast.

5 Conclusion

SCUR is a label-free risk score developed for assessing individual LLM judge-panel decisions. A cross-fitted latent measurement model provides three risk signals: preference uncertainty, measurement cross-entropy, and perturbation sensitivity. Across five benchmarks, SCUR improves error ranking and selective acceptance, identifies errors that remain difficult to detect under high panel consensus, and is robust to different aggregation rules. These signals further provide interpretable evidence for why individual decisions may warrant human review. Together, our results show that agreement alone does not guarantee decision reliability and that SCUR provides a valuable tool for prioritizing decisions for automatic acceptance or human review.

AI USE STATEMENT

Generative AI tools were used to provide feedback on research methodology and experimental design, assist with implementation and debugging of experimental code, support interpretation and presentation of results, and draft and edit manuscript text for clarity and readability. In addition, LLMs were used as judge models in the experiments for the LLMBar and LLMEval^2 benchmarks, as described in the paper. All AI-assisted research decisions and manuscript text were reviewed by the authors, and AI-assisted code and analyses were verified by the authors. The authors take full responsibility for the methodology, experiments, results, claims, and final content of the paper.

ETHICS STATEMENT

SCUR is intended to support, rather than replace, human judgment. A low SCUR score should not be interpreted as a guarantee of correctness, particularly when multiple LLM judges share systematic errors or biases. The framework assesses relative decision risk rather than establishing factual correctness. In high-stakes settings, its scores should be used as evidence for prioritizing human review rather than as a basis for fully automated decisions.

REPRODUCIBILITY STATEMENT

The code and data used in our experiments are available at `https://anonymous.4open.science/r/scur-reproduction-68B0/`.

REFERENCES

Automotive Industry Action Group. *Measurement Systems Analysis Reference Manual*. Automotive Industry Action Group, 4th edition, 2010. URL https://webstore.ansi.org/standards/aiag/aiagmsa2010.

Yang Ba, Michelle V. Mancenido, Erin K. Chiou, and Rong Pan. Data quality in crowdsourcing and spamming behavior detection. *Behavior Research Methods*, 57(9):251, 2025. doi: 10.3758/s13428-025-02757-5. URL https://doi.org/10.3758/s13428-025-02757-5.

Sher Badshah, Ali Emami, and Hassan Sajjad. Judge, retrieve, or abstain: Uncertainty-guarded LLM judging with provable risk guarantees. *arXiv preprint arXiv:2608.17994*, 2026a. URL https://arxiv.org/abs/2608.17994.

Sher Badshah, Ali Emami, and Hassan Sajjad. SCOPE: Selective conformal optimized pairwise LLM judging. *arXiv preprint arXiv:2602.13110*, 2026b. URL https://arxiv.org/abs/2602.13110.

Ralph Allan Bradley and Milton E. Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952. URL https://academic.oup.com/biomet/article-abstract/39/3-4/324/326091.

Chi-Min Chan, Weize Chen, Yusheng Su, Jianxuan Yu, Wei Xue, Shanghang Zhang, Jie Fu, and Zhiyuan Liu. ChatEval: Towards better LLM-based evaluators through multi-agent debate. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2308.07201.

Junhyuk Choi, Sohhyung Park, Chanhee Cho, Hyeonchu Park, and Bugeun Kim. Diagnosing the reliability of LLM-as-a-judge via item response theory. *arXiv preprint arXiv:2602.00521*, 2026. URL https://arxiv.org/abs/2602.00521.

Ido Dagan and Sean P. Engelson. Committee-based sampling for training probabilistic classifiers. In *Machine Learning Proceedings 1995*, pp. 150–157. Morgan Kaufmann, 1995. doi: 10.1016/B978-1-55860-377-6.50027-X. URL https://www.sciencedirect.com/science/article/abs/pii/B978155860377650027X.

A. P. Dawid and A. M. Skene. Maximum likelihood estimation of observer error-rates using the EM algorithm. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 28(1):20–28, 1979. URL https://academic.oup.com/jrsssc/article/28/1/20/6953573.

Yicheng Gao, Gonghan Xu, Zhe Wang, and Arman Cohan. Bayesian calibration of win rate estimation with LLM evaluators. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 4757–4769, Miami, Florida, USA, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.273. URL https://aclanthology.org/2024.emnlp-main.273/.

Yonatan Geifman, Guy Uziel, and Ran El-Yaniv. Bias-reduced uncertainty estimation for deep neural classifiers. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=SJfb5jCqKm.

Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *International Conference on Learning Representations*, 2017. URL https://arxiv.org/abs/1610.02136.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021. URL https://arxiv.org/abs/2009.03300.

Dirk Hovy, Taylor Berg-Kirkpatrick, Ashish Vaswani, and Eduard Hovy. Learning whom to trust with MACE. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 1120–1130, 2013. URL https://aclanthology.org/N13-1132/.

Seungone Kim, Jamin Shin, Yejin Cho, Joel Jang, Shayne Longpre, Hwaran Lee, Sangdoo Yun, Seongjin Shin, Sungdong Kim, James Thorne, and Minjoon Seo. Prometheus: Inducing fine-grained evaluation capability in language models. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2310.08491.

Dawei Li, Bohan Jiang, Liangjie Huang, Alimohammad Beigi, Chengshuai Zhao, Zhen Tan, Amrita Bhattacharjee, Yuxuan Jiang, Canyu Chen, Tianhao Wu, Kai Shu, Lu Cheng, and Huan Liu. 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*, pp. 2757–2791, Suzhou, China, 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.emnlp-main.138. URL https://aclanthology.org/2025.emnlp-main.138/.

Dawei Li, Renliang Sun, Yue Huang, Ming Zhong, Bohan Jiang, Jiawei Han, Xiangliang Zhang, Wei Wang, and Huan Liu. Preference leakage: A contamination problem in LLM-as-a-judge. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=grIvSXVJ65.

Junyi Li, Xiaoxue Cheng, Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. HaluEval: A large-scale hallucination evaluation benchmark for large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 6449–6464, Singapore, 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.397. URL https://aclanthology.org/2023.emnlp-main.397/.

Yen-Ting Lin and Yun-Nung Chen. LLM-eval: Unified multi-dimensional automatic evaluation for open-domain conversations with large language models. In *Proceedings of the 5th Workshop on NLP for Conversational AI (NLP4ConvAI 2023)*, pp. 47–58, 2023. URL https://aclanthology.org/2023.nlp4convai-1.5/.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. G-eval: NLG evaluation using GPT-4 with better human alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 2511–2522, Singapore, 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.153. URL https://aclanthology.org/2023.emnlp-main.153/.

Frederic M. Lord. *Applications of Item Response Theory to Practical Testing Problems*. Lawrence Erlbaum Associates, 1980. URL https://www.ets.org/research/policy_research_reports/publications/book/1980/jexj.html.

R. Duncan Luce. *Individual Choice Behavior: A Theoretical Analysis*. Wiley, 1959. URL https://books.google.com/books/about/Individual_Choice_Behavior.html?id=a80DAQAAIAAJ.

Silviu Paun, Bob Carpenter, Jon Chamberlain, Dirk Hovy, Udo Kruschwitz, and Massimo Poesio. Comparing bayesian models of annotation. *Transactions of the Association for Computational Linguistics*, 6:571–585, 2018. doi: 10.1162/tacl_a_00040. URL https://aclanthology.org/Q18-1040/.

Bhaktipriya Radharapu, Eshika Saxena, Kenneth Li, Chenxi Whitehouse, Adina Williams, and Nicola Cancedda. Calibrating LLM judges: Linear probes for fast and reliable uncertainty estimation. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 6: Industry Track)*, pp. 193–216, San Diego, California, USA, 2026. Association for Computational Linguistics. doi: 10.18653/v1/2026.acl-industry.14. URL https://aclanthology.org/2026.acl-industry.14/.

Kayla Schroeder and Zach Wood-Doughty. Can you trust LLM judgments? reliability of LLM-as-a-judge. *arXiv preprint arXiv:2412.12509*, 2024. URL https://arxiv.org/abs/2412.12509.

Burr Settles. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2009. URL https://research.cs.wisc.edu/techreports/viewreport.php?report=1648.

Lin Shi, Chiyu Ma, Wenhua Liang, Xingjian Diao, Weicheng Ma, and Soroush Vosoughi. 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*, pp. 292–314, Mumbai, India, 2025. The Asian Federation of Natural Language Processing and The Association for Computational Linguistics. doi: 10.18653/v1/2025.ijcnlp-long.18. URL https://aclanthology.org/2025.ijcnlp-long.18/.

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

Peiyi Wang, Lei Li, Liang Chen, Zefan Cai, Dawei Zhu, Binghuai Lin, Yunbo Cao, Lingpeng Kong, Qi Liu, Tianyu Liu, and Zhifang Sui. Large language models are not fair evaluators. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 9440–9450, Bangkok, Thailand, 2024a. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.511. URL https://aclanthology.org/2024.acl-long.511/.

Yidong Wang, Zhuohao Yu, Wenjin Yao, Zhengran Zeng, Linyi Yang, Cunxiang Wang, Hao Chen, Chaoya Jiang, Rui Xie, Jindong Wang, Xing Xie, Wei Ye, Shikun Zhang, and Yue Zhang. PandaLM: An automatic evaluation benchmark for LLM instruction tuning optimization. In *International Conference on Learning Representations*, 2024b. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/be3b0d51a2b86cb4ffe50f13480217e0-Abstract-Conference.html.

Larry Wasserman. *All of Statistics: A Concise Course in Statistical Inference*. Springer, 2004. URL https://link.springer.com/book/10.1007/978-0-387-21736-9.

Koki Wataoka, Tsubasa Takahashi, and Ryokan Ri. Self-preference bias in LLM-as-a-judge. *arXiv preprint arXiv:2410.21819*, 2024. URL https://arxiv.org/abs/2410.21819.

Jacob Whitehill, Paul Ruvolo, Tingfan Wu, Jacob Bergsma, and Javier Movellan. Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. In Y. Bengio, D. Schuurmans, J. Lafferty, C. Williams, and A. Culotta (eds.), *Advances in Neural Information Processing Systems*, volume 22. Curran Associates, Inc., 2009. URL https://proceedings.neurips.cc/paper_files/paper/2009/file/f899139df5e1059396431415e770c6dd-Paper.pdf.

Zhiyuan Zeng, Jiatong Yu, Tianyu Gao, Yu Meng, Tanya Goyal, and Danqi Chen. Evaluating large language models at evaluating instruction following. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2310.07641.

Xinghua Zhang, Bowen Yu, Haiyang Yu, Yangyu Lv, Tingwen Liu, Fei Huang, Hongbo Xu, and Yongbin Li. Wider and deeper LLM networks are fairer LLM evaluators. *arXiv preprint arXiv:2308.01862*, 2023. URL https://arxiv.org/abs/2308.01862.

Jitian Zhao, Changho Shin, Tzu-Heng Huang, Satya Sai Srinath Namburi GNVV, and Frederic Sala. CARE: Confounder-aware aggregation for reliable LLM evaluation. *arXiv preprint arXiv:2603.00039*, 2026. URL https://arxiv.org/abs/2603.00039.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging LLM-as-a-judge with MT-bench and chatbot arena. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://arxiv.org/abs/2306.05685.

Han Zhou, Jordy Van Landeghem, Teodora Popordanoska, and Matthew B. Blaschko. A novel characterization of the population area under the risk coverage curve (AURC) and rates of finite sample estimators. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 79226–79253. PMLR, 2025. URL https://proceedings.mlr.press/v267/zhou25y.html.

A Model Structure and Estimation

This appendix specifies the latent measurement model and the estimator used for the reported experiments. The latent item margin describes a preference shared by the judge panel. Truth labels are used only in evaluation.

A.1 Measurement Model and Parameter Interpretation

The measurement model in Equation 1 is

$$\operatorname{logit}(p^{\mathrm{llm}}_{ijc}) = a_j q_i + b_j + \delta_c + \kappa_{jc} + \epsilon_{ijc}, \qquad a_j > 0, \qquad \epsilon_{ijc} \sim \mathcal{N}(0, \sigma_j^2).$$

The model is fitted to log-odds. Judge verdicts are retained for majority voting and the vote-based baselines. Each non-original condition is a categorical intervention; Appendix C.2 lists the conditions available for each benchmark.

Table 3: Parameters in the fitted continuous-response measurement model.

Parameter	Interpretation
q_i	Direction and strength of the shared item preference.
a_j	Positive discrimination: how strongly judge j responds to q_i.
b_j	Judge-specific tendency toward canonical candidate A.
δ_c	Shared shift under condition c, relative to the original condition.
κ_{jc}	Judge-specific deviation from the shared condition shift.
σ_j	Residual log-odds scale for judge j, estimated after fitting the mean structure.

When only original-condition responses are available, δ_c and κ_{jc} are absent and the model reduces to $\operatorname{logit}(p^{\mathrm{llm}}_{ij0}) = a_j q_i + b_j + \epsilon_{ij0}$.

A.2 Shrinkage and Coordinate Conventions

The released estimator uses ridge least-squares updates for the mean structure. Its blockwise quadratic penalties are

$$\mathcal{R} = \frac{\lambda_q}{2}\sum_i q_i^2 + \frac{\lambda_a}{2}\sum_j (a_j - 1)^2 + \frac{\lambda_b}{2}\sum_j b_j^2 + \frac{\lambda_\delta}{2}\sum_{c\neq 0}\delta_c^2 + \frac{\lambda_\kappa}{2}\sum_j\sum_{c\neq 0}\kappa_{jc}^2.$$

The strengths are $\lambda_q = \lambda_a = \lambda_b = 1$, $\lambda_\delta = 5$, and $\lambda_\kappa = 10$. These penalties correspond to Gaussian shrinkage toward zero for q, b, δ, κ and toward one for a, with positivity enforced for a. The model separates shared item and condition components from judge-specific responses and condition deviations.

The original condition is fixed by $\delta_0 = \kappa_{j0} = 0$. To fix the multiplicative scale, after each discrimination update the implementation sets $g = \exp[J^{-1}\sum_j \log a_j]$ and replaces $a_j \leftarrow a_j/g$ and $q_i \leftarrow g q_i$, preserving $a_j q_i$. For each non-original condition, it transfers the mean of $\kappa_{\cdot c}$ into δ_c, so that $\sum_j \kappa_{jc} = 0$ while preserving the total condition effect.

A.3 Optimization and Held-Out Item Inference

Let $m_{ijc} = \mathrm{logit}(p_{ijc}^{\mathrm{llm}})$. Each update regresses the residual left by the other blocks on the current block. For example, with $\mathcal{O}_i$ denoting the observed judge–condition cells for a fit item,

$$q_i \leftarrow \frac{\sum_{(j,c)\in\mathcal{O}_i} a_j(m_{ijc} - b_j - \delta_c - \kappa_{jc})}{\lambda_q + \sum_{(j,c)\in\mathcal{O}_i} a_j^2}.$$

Before scale normalization, each discrimination update is clipped to $[0.05, 20]$. The mean-structure fit uses equal observation weights. After fitting, the residual scale is estimated as

$$\widehat{\sigma}_j^2 = \frac{1}{n_j}\sum_{(i,c)\in\mathcal{O}_j} (m_{ijc} - \widehat{a}_j\widehat{q}_i - \widehat{b}_j - \widehat{\delta}_c - \widehat{\kappa}_{jc})^2 + 10^{-9}.$$

These residual estimates do not reweight the fitted mean structure. Here $\mathcal{O}_j$ contains the observed item–condition cells for judge j, and $n_j = |\mathcal{O}_j|$.

We use three random restarts, a nominal 700 iterations, and an automatic extension up to 2,100 iterations. The standard stopping rule requires the root mean square change in predicted margins to be below 10^{-6}, the maximum change below 10^{-5}, and the relative change in the objective below 10^{-7} for three successive iterations. Among converged restarts, the one with the smallest fit objective is retained. The criterion is mean squared residual plus the implementation's item- and judge-bias ridge diagnostic; it is used for convergence and restart selection.

For each seed, items are randomly partitioned into fit, reference, and evaluation splits in a 50/30/20 ratio. Only fit items estimate the judge and perturbation condition parameters. For a reference or evaluation item, the leave-one-judge-out margin is obtained using all available conditions from judges other than j:

$$q_i^{(-j)} = \frac{\sum_{(\ell,c)\in\mathcal{O}_i:\ell\neq j} \widehat{a}_\ell(m_{i\ell c} - \widehat{b}_\ell - \widehat{\delta}_c - \widehat{\kappa}_{\ell c})}{\lambda_q + \sum_{(\ell,c)\in\mathcal{O}_i:\ell\neq j} \widehat{a}_\ell^2}.$$

The corresponding held-out prediction is $\widehat{p}_{ijc} = \sigma(\widehat{a}_j q_i^{(-j)} + \widehat{b}_j + \widehat{\delta}_c + \widehat{\kappa}_{jc})$. Excluding every condition of the held-out judge prevents its own responses from determining the item margin used to assess them. Reference items supply the signal ECDFs, which are frozen before evaluation. Neither model fitting nor reference normalization uses ground-truth labels. For numerical stability, the score implementation floors each complement $1 - z_i^k$ at 10^{-12}. A sensitivity component that is absent or constant on the reference split is omitted; the maximum uses the remaining sensitivity component, if any.

B Risk Baselines

We compare SCUR with vote entropy (VE), predictive entropy (PE), Dawid–Skene posterior entropy (DS entropy), and mean per-judge MSP uncertainty. All four use original-condition responses ($c = 0$), with larger scores indicating greater uncertainty.

B.1 Common Inputs and Evaluation Protocol

We use the notation of the main text: J_i is the number of available judges for item i, and p^{llm}_{ij0} is judge j's probability that the original candidate A_i wins. Let $y_{ij0} \in \{0, 1\}$ denote its binary verdict for that same candidate, with one indicating A_i. The sums below run over the J_i available judges.

We use the binary entropy function H defined in Section 3.2, with base-2 logarithms for the entropy baselines. PE, VE, and MSP require no fitted parameters. DS is fitted on the unlabeled $\mathcal{D}_{\mathrm{fit}}$ and then held fixed on $\mathcal{D}_{\mathrm{eval}}$. Gold labels are used only for evaluation.

B.2 Predictive Entropy

Predictive entropy of verdict-token probabilities has been used to guide abstention by LLM judges (Badshah et al., 2026a). We apply it to the panel by averaging the LLM probabilities before taking entropy:

$$\overline{p}^{\mathrm{llm}}_i = \frac{1}{J_i} \sum_{j=1}^{J_i} p^{\mathrm{llm}}_{ij0}, \qquad \mathrm{PE}_i = H(\overline{p}^{\mathrm{llm}}_i).$$

The superscript distinguishes this mean from $\overline{p}_i$ in the main text, which averages cross-fitted LMM predictions and is used to compute U_i. PE uses the LLM probabilities directly, without fitting the LMM.

PE measures uncertainty in the panel-mean preference. It is largest when the mean is 0.5 and small when the mean is near zero or one. A mean near 0.5 can reflect individually uncertain judges or confident judges favoring opposite answers. Its range is $[0, 1]$.

B.3 Vote Entropy

VE uses the binary verdicts rather than their probabilities:

$$\mathrm{VE}_i = H\left(\frac{1}{J_i} \sum_{j=1}^{J_i} y_{ij0} \right).$$

It is zero for unanimous votes and largest for an evenly divided panel. Items with the same vote split have the same VE, even when their LLM probabilities differ. Thus both unanimously correct and unanimously incorrect panels receive zero VE. Its range is $[0, 1]$.

Vote entropy is a committee-disagreement measure (Dagan & Engelson, 1995; Settles, 2009). Recent LLM-judge calibration work evaluates majority confidence across repeated verdicts (Radharapu et al., 2026). For binary verdicts, VE is a monotonic transformation of one minus majority confidence.

B.4 Dawid–Skene Posterior Entropy

A Bayesian variant of the Dawid–Skene model (Dawid & Skene, 1979) has been applied to LLM-evaluator calibration (Gao et al., 2024), while DS is used as an aggregation baseline in CARE (Zhao et al., 2026). We use entropy of its posterior as a risk score, extending these aggregation applications to error ranking.

DS estimates a class prior and a confusion matrix for each judge from original-condition binary verdicts, assuming judges are conditionally independent given an item's unobserved label. Let $\widehat{\rho}_i$ be its fitted posterior probability that candidate A_i is preferred, given $y_{i10}, \ldots, y_{iJ_i0}$. Then

$$\mathrm{DS\ entropy}_i = H(\widehat{\rho}_i).$$

This DS posterior is distinct from both the observed LLM probabilities p_{ij0}^{llm} and the cross-fitted LMM predictions $\widehat{p}_{ij0}$.

We fit the prior and confusion matrices by expectation–maximization on $\mathcal{D}_{\mathrm{fit}}$, with additive smoothing of 0.5. Initialization uses the proportion of votes for A_i, clipped to $[10^{-3}, 1-10^{-3}]$. Estimation stops after 200 iterations or when the maximum change in posterior probabilities is below 10^{-7}. The label orientation is chosen by agreement with fit-set majority votes, without gold labels; swapping the orientation does not change entropy. The fitted parameters remain fixed for evaluation.

DS entropy measures uncertainty after accounting for learned judge confusion patterns. It uses binary votes, not p_{ij0}^{llm}. A confidently wrong posterior can still have low entropy. Its range is $[0, 1]$. As in the main text, DS entropy ranks items by risk; DS aggregation instead uses the posterior to choose their final verdicts.

B.5 Mean Per-Judge MSP Uncertainty

Maximum softmax probability (MSP) is a confidence measure (Hendrycks & Gimpel, 2017), also evaluated for selective prediction with LLM judges (Radharapu et al., 2026). Our risk baseline averages one minus this confidence across judges:

$$\mathrm{MSP}_i = \frac{1}{J_i} \sum_{j=1}^{J_i} \left[1 - \max\left(p_{ij0}^{\mathrm{llm}}, 1 - p_{ij0}^{\mathrm{llm}}\right)\right].$$

The MSP baseline reports uncertainty: larger values indicate more uncertain judges. Its range is $[0, 0.5]$. It remains low when every judge is confident, including when confident judges disagree. The maximum is taken separately for each judge *before* averaging. Taking $1 - \max(\overline{p}_i^{\mathrm{llm}}, 1 - \overline{p}_i^{\mathrm{llm}})$ instead would measure uncertainty of the panel mean and produce the same ranking as binary PE.

Comparability and ties. All four baselines are evaluated against the same e_i^{MV} defined in the main text, on the same held-out items. AUROC uses average ranks for equal scores. For $1 - \mathrm{AURC}$ and Err@50, we average over uniform random orderings within tied scores, so file order cannot affect selective acceptance results. Different raw score ranges do not affect these ranking-based comparisons.

C Benchmark Details and Judging Protocol

C.1 Benchmark Construction and Validation Labels

Chatbot-Arena. A pairwise-preference benchmark contains $17{,}325$ items, 17 judge models, and $589{,}050$ judge observations. Each seed has $5{,}198$ reference items and $3{,}465$ evaluation items. The panel-error target compares the original-condition majority decision with the human pairwise-preference label.

HaluEval. A hallucination-detection benchmark contains $20{,}000$ items, 6 judges, and $120{,}000$ observations. Each seed has $6{,}000$ reference items and $4{,}000$ evaluation items. The benchmark label determines whether the original-condition panel verdict is correct.

MMLU. A multiple-choice benchmark contains $28{,}084$ items, 17 judges, and $477{,}428$ observations. Each seed has $8{,}425$ reference items and $5{,}617$ evaluation items. The objective answer key supplies the validation label.

LLMBar. An instruction-following benchmark contains 419 items, 5 judges, and $12{,}570$ observations across 6 conditions. Each seed has 126 reference items and 83 evaluation items. The expert preference label determines whether the original-condition panel verdict is correct.

LLMEval^2. We use a 200-item subset of the pairwise-preference benchmark, evaluated by 5 judges, yielding $6{,}000$ observations across 6 conditions. Each seed has 60 reference items and 40 evaluation items. The source preference label determines whether the original-condition panel verdict is correct.

C.2 Perturbations Available for Each Benchmark

A perturbation evaluates the same task and candidate responses under a modified answer order or judging criterion. Table 4 summarizes the available conditions.

Table 4: Judging conditions and availability of perturbation sensitivity.

Benchmark	Conditions	S available?
Chatbot-Arena	Original and swapped answer orders.	Yes
HaluEval / MMLU	Original condition only.	No
LLMBar	Original/swapped answer order crossed with standard/strict/evidence-first rubric.	Yes
LLMEval^2	The same two-order, three-rubric design.	Yes

Answer-order swap. The two candidate responses exchange their displayed positions while the task and rubric remain fixed. The resulting probabilities and verdict votes are expressed relative to the original candidate A_i: after a swap, a probability p of choosing displayed Response A corresponds to probability $1 - p$ of choosing A_i.

Rubric changes and paired comparisons. For LLMBar and LLMEval^2, each answer order is crossed with three rubrics, described below, giving six conditions. Original order with the standard rubric defines $c = 0$. Each of the other five conditions is paired with this original condition for the same item and judge; "paired" thus refers to comparing judgments of the same candidate pair, not different items. These comparisons include order changes, rubric changes, and their combinations. The task and candidate texts remain fixed.

C.3 Judge Panel

LLMBar and LLMEval^2 use the same five judge models:

- `olmo3-7b-instruct`
- `qwen3-30b-a3b-instruct-2507`
- `llama4-scout-17b`
- `granite41-30b`
- `llama3-groq-70b-tool-use`

Every judge evaluates every item under all six conditions, yielding 30 judgments per item. Each judgment uses the shared system and task prompts below, with temperature zero and a one-token A/B output. The probability p^{llm}_{ijc} is obtained from verdict-token log-probabilities and expressed relative to the original candidate A_i, as in the main text.

Judging criteria. The *standard* rubric asks the judge to choose the better or correct response. The *strict* rubric makes any material factual or instruction-following error grounds for rejecting a response. The *evidence-first* rubric asks the judge to check factual support and instruction compliance internally before choosing. All three return only an A/B verdict. The rubric texts appear in the examples below.

C.4 Prompts and Perturbation Examples

The system message is identical for both benchmarks and all conditions:

```
You are an impartial evaluator. Return exactly one character: A or B.
Do not explain.
```

Each box below contains a complete task prompt. We illustrate the three perturbations with three different benchmark items, showing each perturbed prompt alongside its own original prompt. Within

each pair, the task and candidate texts are unchanged. The original condition always uses the original answer order and the standard rubric. Order swaps can also be combined with either modified rubric, giving the six conditions described above.

Answer-order swap. The task asks for two rhymes for "Moon." The original displays "Soon, Loon" as Response A; the swap displays it as Response B. Only the response positions change, while the task, standard criterion, and displayed-label semantics remain fixed.

Original.

```
Evaluation task:
Generate two rhymes for the following word. Moon

Response A:
Soon, Loon

Response B:
Clone, June, Nore, Tune

Label semantics: A means Response A is better; B means Response B is
better.

Criterion:
Use the standard criterion: choose the better/correct option.

Answer with A or B only.
```

Swapped answer order.

```
Evaluation task:
Generate two rhymes for the following word. Moon

Response A:
Clone, June, Nore, Tune

Response B:
Soon, Loon

Label semantics: A means Response A is better; B means Response B is
better.

Criterion:
Use the standard criterion: choose the better/correct option.

Answer with A or B only.
```

Strict rubric. This task asks for an eight-character password. The candidates contain ten and eight characters, respectively. The strict condition replaces the standard criterion with an explicit instruction to reject material factual or instruction-following errors; it leaves the task, candidates, and their order unchanged.

Original.

```
Evaluation task:
Generate a computer password that is 8 characters long

Response A:
hx3eLXPaPu

Response B:
X7#hklzy

Label semantics: A means Response A is better; B means Response B is
better.

Criterion:
Use the standard criterion: choose the better/correct option.

Answer with A or B only.
```

Strict rubric.

```
Evaluation task:
Generate a computer password that is 8 characters long

Response A:
hx3eLXPaPu

Response B:
X7#hklzy

Label semantics: A means Response A is better; B means Response B is
better.

Criterion:
Apply the criterion strictly. A response with any material factual or
instruction-following error should lose.

Answer with A or B only.
```

Evidence-first rubric. The task asks for a description of structured restaurant data specifying a cheap price range. One candidate also calls the restaurant expensive. The evidence-first condition replaces only the criterion, asking the judge to check factual support and instruction compliance internally before choosing. No external evidence is supplied and no written explanation is requested; the output remains a single A/B verdict.

Original.

```
Evaluation task:
Describe the given structural data:

inform(name='X',type=placetoeat,eattype=restaurant,pricerange=cheap,
near='X')

Response A:
X is a cheap restaurant in the expensive price range near X .

Response B:
X is a cheap restaurant near X .

Label semantics: A means Response A is better; B means Response B is
better.

Criterion:
Use the standard criterion: choose the better/correct option.

Answer with A or B only.
```

Evidence-first rubric.

```
Evaluation task:
Describe the given structural data:

inform(name='X',type=placetoeat,eattype=restaurant,pricerange=cheap,
near='X')

Response A:
X is a cheap restaurant in the expensive price range near X .

Response B:
X is a cheap restaurant near X .

Label semantics: A means Response A is better; B means Response B is
better.

Criterion:
First check factual support and instruction compliance internally, then
choose the better/correct option.

Answer with A or B only.
```

D HIGH-CONSENSUS ANALYSIS

Figure 6 complements the high-consensus risk analysis in Figure 4 in two settings: comparing items with the same vote pattern, and evaluating items with high panel-mean confidence.

Exact-matched analysis. We compare items with the same vote pattern under the original condition: each judge chooses the same candidate across the compared items. This keeps both the vote split and the judges supporting each candidate fixed. We compute AUROC within each group containing both incorrect and correct decisions, then average these values using the number of incorrect–

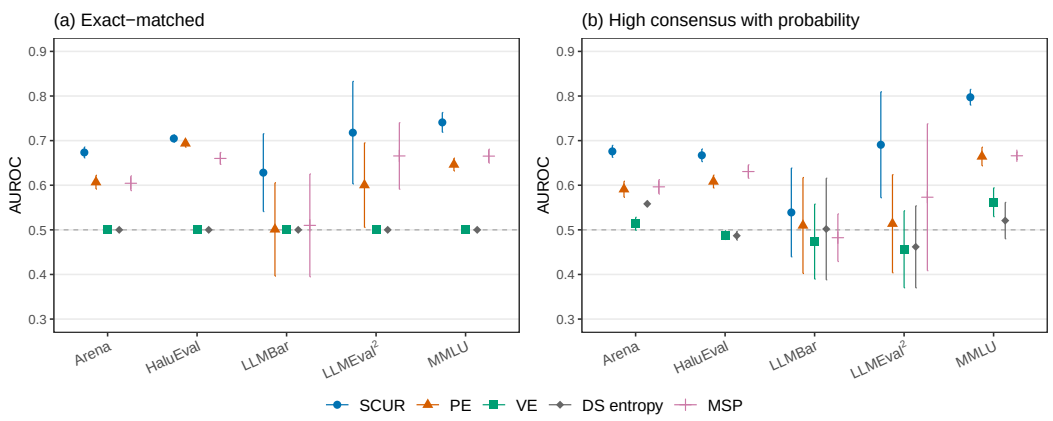

Figure 6: **Error ranking under exact matching and high consensus with probability.** (a) AUROC comparing items with the same vote pattern; VE and DS entropy are .5. (b) AUROC among items with panel-mean confidence at least .8. Both panels compare SCUR with the same four baselines. Points and whiskers show means and ± 1 sample standard deviation over five splits. Dashed lines mark chance performance.

correct item pairs in each group as weights:

$$\mathrm{AUROC}_{\mathrm{pattern}} = \frac{\sum_g n_g^+ n_g^- \, \mathrm{AUROC}_g}{\sum_g n_g^+ n_g^-},$$

where n_g^+ and n_g^- count incorrect and correct decisions in group g. VE and fitted DS entropy are constant within each group, giving AUROC .5. SCUR's average AUROC is .693, versus .610 for PE and .621 for MSP (panel a). Thus SCUR can distinguish incorrect from correct decisions even when the judges make the same choices.

High consensus with probability. Using the panel-mean probability $\overline{p}_i^{\mathrm{llm}}$ defined in Appendix B, high consensus with probability selects items satisfying $\max(\overline{p}_i^{\mathrm{llm}}, 1-\overline{p}_i^{\mathrm{llm}}) \geq .8$. This is equivalent to $\mathrm{PE}_i \leq H(.8)$: the test asks whether errors remain distinguishable within a low-PE subset.

SCUR has the highest mean AUROC among the five methods on all benchmarks in panel (b). On Arena, its AUROC is .676, compared with .596 for the best baseline, MSP. This supports detecting residual risk even when the panel's mean probability strongly favors one answer. The LLMBar AUROC remains near chance (.539), and LLMEval^2 has only 3–5 errors per split, limiting the precision of its estimate. The two panels examine complementary aspects of high consensus, showing that strong judge agreement does not eliminate meaningful variation in error risk.