

How to Correctly Report LLM-as-a-Judge Evaluations

Chungpa Lee¹ Thomas Zeng² Jongwon Jeong² Jy-yong Sohn¹ Kangwook Lee^{2,3}

Abstract

Large language models (LLMs) are widely used as scalable evaluators of model responses in lieu of human annotators. However, imperfect sensitivity and specificity of LLM judgments induce bias in naive evaluation scores. We propose a simple plug-in framework that corrects this bias and constructs confidence intervals accounting for uncertainty from both the test dataset and a human-evaluated calibration dataset, enabling statistically sound and practical LLM-based evaluation. Building on this framework, we introduce an adaptive calibration strategy for constructing the calibration dataset to reduce uncertainty in the estimated score. Notably, we characterize the regimes in which LLM-based evaluation within our framework produces more reliable estimates than fully human evaluation. Moreover, our framework is more robust to distribution shift between the test and calibration datasets than existing approaches.

1. Introduction

The use of large language models (LLMs) as judges provides a cheap, scalable alternative to human evaluation for various tasks like grading factual accuracy, assessing code quality or detecting harmful content (Zheng et al., 2023; Liu et al., 2023; Wang et al., 2023; Li et al., 2025; Gu et al., 2025). A common practice is to report the proportion of responses that an LLM judges as ‘correct’, denoted by \hat{p} . However, directly reporting raw LLM judgment scores is statistically problematic (Bross, 1954; Schwartz, 1985; Forman, 2005; Angelopoulos et al., 2023a; Boyeau et al., 2025; Fraser, 2024; Albinet, 2025). Because LLM judgments are inherently noisy, \hat{p} generally deviates from the true accuracy and fails to provide a reliable estimate (Wang et al., 2024; Koo et al., 2024; Huang et al., 2025).

To understand the source of the resulting bias in the judg-

¹Yonsei University, Seoul, Korea ²University of Wisconsin–Madison, Wisconsin, USA ³KRAFTON, Seoul, Korea. Correspondence to: Jy-yong Sohn <jysohn1108@yonsei.ac.kr>, Kangwook Lee <kangwook.lee@wisc.edu>.

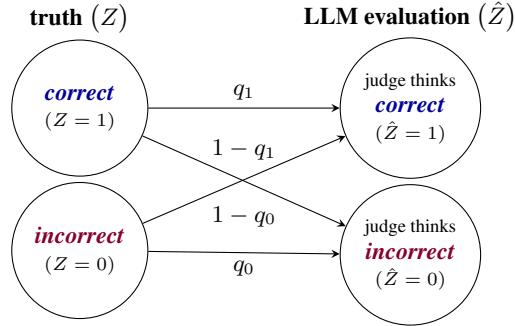


Figure 1. LLM judgments with error rates $1 - q_1$ and $1 - q_0$, where q_1 and q_0 are LLM’s sensitivity and specificity, respectively.

ment score \hat{p} , we first examine how an LLM judge makes errors when evaluating individual responses. As illustrated in Figure 1, an LLM may incorrectly label an ‘incorrect’ response as ‘correct’ or, conversely, mislabel a ‘correct’ response as ‘incorrect’. Let q_1 and q_0 denote the probabilities that the LLM correctly judges ‘correct’ and ‘incorrect’ responses, respectively. These quantities correspond to the sensitivity q_1 and specificity q_0 of the LLM judge.

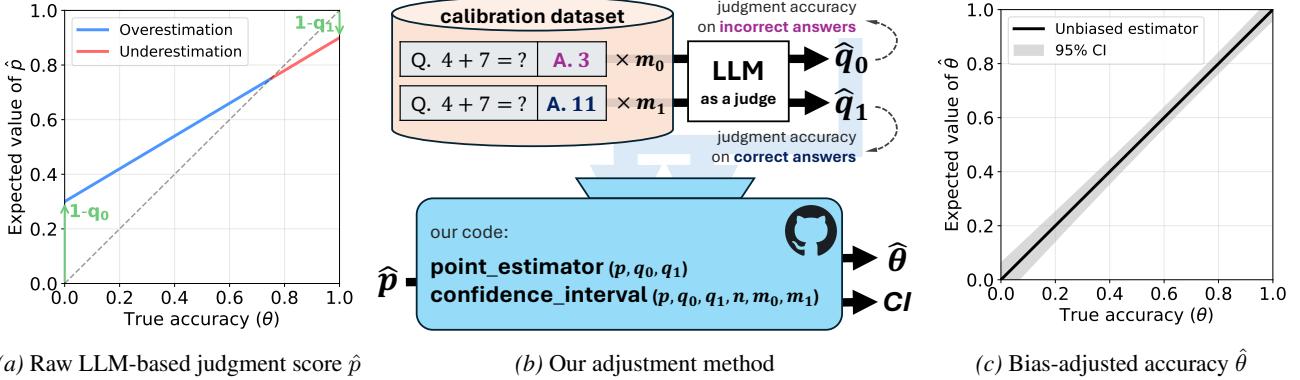
In extreme cases, such as when $q_1 = 1$ and $q_0 = 0$, the LLM judges every response as ‘correct’. Consequently, the average of the LLM’s binary judgments \hat{p} deviates from the true accuracy θ , becoming equal to 1 regardless of θ . This illustrates that the raw judgment score can be biased.

In general, whenever the LLM is imperfect ($q_0 + q_1 < 2$), the expected value of \hat{p} deviates from the true accuracy θ :

$$\mathbb{E}[\hat{p}] = \theta + (2 - q_0 - q_1) \left(\frac{1 - q_0}{2 - q_0 - q_1} - \theta \right),$$

implying positive bias at low θ and negative bias at high θ (see Section 5 for details). This is illustrated in Figure 2a for an LLM with $q_1 = 0.9$ and $q_0 = 0.7$. $\mathbb{E}[\hat{p}]$ overestimates θ when $\theta < 0.75$ (blue line) and underestimates it when $\theta > 0.75$ (red line), which arises from the two underlying judgment errors (green arrows). A high probability of wrongly rejecting a ‘correct’ response (large $1 - q_1$) induces negative bias at high accuracies, whereas a high error probability of wrongly accepting an ‘incorrect’ response (large $1 - q_0$) induces positive bias at low accuracies.

This issue is not merely theoretical. With the increasing adoption of LLM-based evaluation, reported improvements may be driven by bias induced by judgment errors rather


 (a) Raw LLM-based judgment score \hat{p}

(b) Our adjustment method

 (c) Bias-adjusted accuracy $\hat{\theta}$

Figure 2. Bias and its adjustment in LLM-based judgment under imperfect LLM evaluators ($q_1 = 0.9$ and $q_0 = 0.7$). **(a)** When the true accuracy θ is low ($\theta < 0.75$), the expected value of the raw LLM-based judgment score $\mathbb{E}[\hat{p}]$ overestimates θ , whereas when θ is high ($\theta > 0.75$), it underestimates θ . **(b)** By accounting for the sensitivity q_1 and specificity q_0 of the LLM judge, which can be estimated from a calibration dataset with true labels, we obtain the bias-adjusted estimator $\hat{\theta}$ along with its confidence interval (CI). **(c)** The resulting estimator $\hat{\theta}$ is unbiased when the true values of q_0 and q_1 are known or when a calibration dataset of large size is available. A plug-in Python implementation of this procedure is provided in <https://github.com/UW-Madison-Lee-Lab/LLM-judge-reporting>.

than true model gains. Since different evaluation procedures can induce biases of different magnitude and direction, some apparent advances in the literature may arise from evaluation artifacts rather than genuine improvements. This motivates careful comparisons and the use of calibrated judges when interpreting past findings, and highlights the need for a principled method for bias adjustment.

Fortunately, this bias can be corrected. When the sensitivity q_1 and specificity q_0 are known, a classical result (Rogan & Gladen, 1978) provides an exact adjustment. Even when they are unknown, they can be estimated from a calibration dataset with human-evaluated labels (Buonaccorsi, 2010), and the resulting estimates \hat{q}_1 and \hat{q}_0 can be substituted into the correction formula. This adjustment produces the bias-adjusted estimator $\hat{\theta}$ in Figure 2c by correcting bias.

Correcting this bias, however, is only part of the problem. LLM-as-a-judge evaluation involves two sources of uncertainty: (i) randomness arising from the test dataset, which affects the judgment score \hat{p} , and (ii) randomness from the calibration dataset, which affects \hat{q}_0 and \hat{q}_1 . A principled confidence interval must incorporate both components; yet prior discussion on LLM-as-a-judge evaluation has focused largely on bias (Fraser, 2024; Albinet, 2025), offering no method for constructing valid intervals.

This work provides a statistical framework for LLM-as-a-judge evaluation, as outlined in Figure 2b. Our key contributions are summarized below:

- In Section 4, we introduce an estimator for the true accuracy θ that corrects bias in LLM-based judgments and derive confidence intervals that account for uncertainty from both the test and calibration datasets. We further propose an adaptive strategy for allocating calibration samples that reduces confidence interval length.

- In Section 6, we characterize the regimes in which LLM-as-a-judge evaluation within our framework yields lower-variance estimates of θ than direct human evaluation. In Section 7, we show that our framework is more robust to distribution shift between the test and calibration datasets than existing approaches.
- In Section 5, we provide theoretical justification for our method, and in Section 8, we validate this theory through extensive Monte Carlo simulations and real-world evaluations on the Chatbot Arena benchmark.

2. Related Work

Statistical Reliability in LLMs’ Evaluation. Recent efforts have sought to formalize the statistical reliability of language model evaluations. While frameworks for calculating error bars and confidence intervals in benchmarks exist (Miller, 2024), they assume that LLM evaluators provide ground-truth labels. However, this assumption does not hold in the LLM-as-a-judge setting, where imperfect sensitivity and specificity introduce bias into LLM-based judgment scores. Recent works have examined these biases across various contexts, including natural language inference and test-time compute scaling (Godbole & Jia, 2025; Mukherjee et al., 2025; Feng et al., 2025). Aligned with this line of work, we propose a bias-adjusted estimator together with statistically sound confidence intervals.

Estimation Methods to Mitigate Bias. To address bias resulting from imperfect verifiers (e.g., diagnostic or screening tests, or LLM-based judges), Rogan & Gladen (1978) established an adjustment to correct bias in prevalence estimates given known sensitivity and specificity. This line of work has been extended by more general frameworks such as

Prediction-Powered Inference (Angelopoulos et al., 2023a;b; Zrnic & Candès, 2024; Broska et al., 2025; Boyneau et al., 2025), which estimates and corrects bias using a dataset with true labels, as well as calibration-based estimator methods (Buonaccorsi, 2010; Kloos et al., 2021; Meertens et al., 2022), as detail in Section 7. While each approach offers distinct advantages, our method builds on Rogan & Gladen (1978); Lang & Reiczigel (2014), adopting distributional assumptions that are robust to shifts between calibration and test datasets and well-suited to the LLM-as-a-judge setting.

3. Problem Setup: LLM-as-a-Judge

Evaluation Objective. We consider the problem of evaluating responses using human judgment. For example, each evaluation instance may consist of a response produced by a given model¹, together with the corresponding input, such as a question. We assume that humans can assess whether a given response is ‘correct’ or ‘incorrect’. This assessment is formalized by a ground-truth labeling function $z : \mathcal{X} \rightarrow \{0, 1\}$, where $z(x) = 1$ indicates that humans judge the response in instance x to be ‘correct’, and $z(x) = 0$ otherwise. Applying this function to a random instance X induces a binary random variable $Z := z(X)$.

Our goal is to estimate the true accuracy of the responses with respect to human judgment, defined as

$$\theta := \Pr(Z = 1) = \mathbb{E}[Z]. \quad (1)$$

Test Distribution and LLM-Based Judgment. Let \mathbb{P} denote the distribution over test instances to be evaluated. In practice, instead of relying on human annotators, an LLM is used as a surrogate judge. Let $\hat{Z} := f_{\text{LLM}}(X) \in \{0, 1\}$ denote the LLM’s judgment, where $\hat{Z} = 1$ indicates that the LLM marks the response as ‘correct’, and $\hat{Z} = 0$ otherwise.

Let $[n] := \{1, \dots, n\}$, where n denotes the test set size. Given a test dataset $\{x_i\}_{i \in [n]}$ sampled i.i.d. from \mathbb{P} , the LLM produces predictions $\hat{z}_i := f_{\text{LLM}}(x_i)$. The quantity typically reported in practice is the empirical fraction of instances judged as ‘correct’ by the LLM:

$$\hat{p} := \frac{1}{n} \sum_{i \in [n]} \hat{z}_i. \quad (2)$$

This estimator targets the probability $p := \Pr_{\mathbb{P}}(\hat{Z} = 1)$ that the LLM judges a randomly drawn test instance as ‘correct’.

Mismatch Between LLM and Human Judgments. The LLM’s judgment \hat{Z} does not necessarily coincide with the human-evaluated true label Z . In particular, the LLM may incorrectly reject responses that are truly ‘correct’ or accept

¹The model producing the response may be an LLM, but rule-based or statistical models are also possible.

responses that are truly ‘incorrect’. These two types of errors are captured by the following parameters:

$$q_1 := \Pr(\hat{Z} = 1 | Z = 1), \quad q_0 := \Pr(\hat{Z} = 0 | Z = 0), \quad (3)$$

which correspond to the *sensitivity* (true positive rate) and *specificity* (true negative rate) of the LLM judge, respectively (Forman, 2008; Lang & Reiczigel, 2014). Unless the LLM is perfectly accurate (i.e., $q_0 = q_1 = 1$), the naive estimator \hat{p} in Eq. (2) is generally a biased estimator of θ .

Calibration Distribution and Estimation of Sensitivity and Specificity. Let \mathbb{Q} denote the distribution over calibration instances. Unlike the test dataset, each calibration instance is evaluated by humans, so that the ground-truth variable Z and the corresponding LLM judgment \hat{Z} are observable.

Let m denote the calibration sample size, and let m_1 and m_0 denote the numbers of calibration instances with $z_j = 1$ and $z_j = 0$, respectively, where $m = m_0 + m_1$. The index j is used to distinguish calibration instances from test instances indexed by i . Using this dataset, we estimate the sensitivity q_1 and specificity q_0 of the LLM judge in Eq. (3) as

$$\begin{aligned} \hat{q}_1 &:= \frac{1}{m_1} \sum_{j \in [m]} \mathbf{1}\{\hat{z}_j = 1, z_j = 1\}, \\ \hat{q}_0 &:= \frac{1}{m_0} \sum_{j \in [m]} \mathbf{1}\{\hat{z}_j = 0, z_j = 0\}, \end{aligned}$$

where $\mathbf{1}\{\cdot\}$ denotes the indicator function. Throughout this work, we allow the distribution of Z under \mathbb{P} and \mathbb{Q} to differ.

Problem Statement. Because the LLM judge is imperfect, the naive estimator \hat{p} generally satisfies $\mathbb{E}[\hat{p}] \neq \theta$, where θ is the true accuracy defined in (1). Moreover, existing LLM-as-a-judge evaluations typically report only this point estimate, without quantifying uncertainty through confidence intervals. As a result, reported accuracies may appear precise even when they are statistically unreliable.

Our objective is therefore twofold: (i) to construct a bias-adjusted estimator of the true accuracy θ , and (ii) to provide statistically sound confidence intervals that reflect uncertainty arising from both the test and calibration datasets.

4. Method to Correctly Report LLM-as-a-Judge Evaluations

In this section, we present a bias-adjusted estimator and a confidence interval for LLM-as-a-judge evaluations. We further analyze how sample sizes affect these intervals and propose an adaptive allocation strategy for constructing the calibration dataset that reduces confidence interval length.

Mitigating Bias on Point Estimator. We begin with the setting in which the sensitivity q_1 and specificity q_0 in Eq. (3) are known. In this case, an unbiased estimator of the true accuracy θ in (1) is given by

$$\hat{\theta} \mid q_0, q_1 = \frac{\hat{p} + q_0 - 1}{q_0 + q_1 - 1}, \quad (4)$$

where derivations in this section are deferred to Section 5.

In realistic settings, these accuracies are unknown and must be estimated from a calibration dataset with human-evaluated labels. Substituting estimates \hat{q}_1 and \hat{q}_0 into (4) gives the bias-adjusted estimator (Rogan & Gladen, 1978):

$$\hat{\theta} = \frac{\hat{p} + \hat{q}_0 - 1}{\hat{q}_0 + \hat{q}_1 - 1}. \quad (5)$$

Uncertainty Quantification via Confidence Interval. To quantify uncertainty in $\hat{\theta}$, we derive a confidence interval for θ that incorporates variance contributions from both the test and calibration dataset (Lang & Reiczigel, 2014):

$$\begin{aligned} & \tilde{\theta} + d\tilde{\theta} \\ & \pm z_\alpha \sqrt{\frac{\frac{\tilde{p}(1-\tilde{p})}{\tilde{n}} + (1-\tilde{\theta})^2 \cdot \frac{\tilde{q}_0(1-\tilde{q}_0)}{\tilde{m}_0} + \tilde{\theta}^2 \cdot \frac{\tilde{q}_1(1-\tilde{q}_1)}{\tilde{m}_1}}{(\tilde{q}_0 + \tilde{q}_1 - 1)^2}}, \end{aligned} \quad (6)$$

where values outside the interval $[0, 1]$ are truncated to 0 or 1. Here, z_α denotes the $(1 - \alpha/2)$ quantile of the standard normal distribution, e.g., $z_{0.05} = 1.96$, and the adjusted quantities are defined as

$$\begin{aligned} \tilde{n} &= n + z_\alpha^2, \quad \tilde{m}_0 = m_0 + 2, \quad \tilde{m}_1 = m_1 + 2, \\ \tilde{p} &= \frac{n \cdot \hat{p} + z_\alpha^2 / 2}{n + z_\alpha^2}, \quad \tilde{q}_0 = \frac{m_0 \cdot \hat{q}_0 + 1}{m_0 + 2}, \end{aligned}$$

$$\tilde{q}_1 = \frac{m_1 \cdot \hat{q}_1 + 1}{m_1 + 2}, \quad \tilde{\theta} = \frac{\tilde{p} + \tilde{q}_0 - 1}{\tilde{q}_0 + \tilde{q}_1 - 1}, \quad (7)$$

$$d\tilde{\theta} = 2z_\alpha^2 \left(-(1-\tilde{\theta}) \cdot \frac{\tilde{q}_0(1-\tilde{q}_0)}{\tilde{m}_0} + \tilde{\theta} \cdot \frac{\tilde{q}_1(1-\tilde{q}_1)}{\tilde{m}_1} \right). \quad (8)$$

Impact of Sample Sizes on Confidence-Interval Length. The confidence interval in (6) reflects uncertainty from both the test and calibration datasets through \tilde{n} , \tilde{m}_0 , and \tilde{m}_1 . As these sample sizes increase, the terms inside the square root decrease, resulting in a shorter confidence interval for θ . Because LLM-as-a-judge evaluations can be run at scale with minimal cost, the test-set size n can often be made extremely large. In the limit $n \rightarrow \infty$, test-set uncertainty vanishes, and the interval length is determined solely by the calibration sample sizes m_0 and m_1 . This observation enables practitioners to target a desired interval length and determine the minimal calibration budget required to achieve it. In

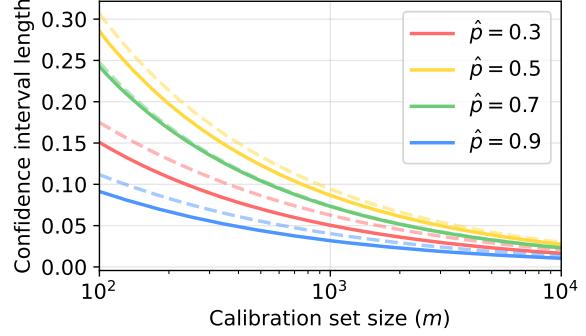


Figure 3. Confidence-interval length across the calibration set size for different values of $\hat{p} \in \{0.3, 0.5, 0.7, 0.9\}$, when $\hat{q}_0 = 0.7$, $\hat{q}_1 = 0.9$, and $n \rightarrow \infty$. Dashed lines correspond to calibration datasets containing equal numbers of ‘incorrect’ and ‘correct’ instances ($m_0 = m_1$), while solid lines correspond to calibration samples allocated using the adaptive rule introduced in Algorithm 1. The allocation reduces the confidence-interval length.

Section 6, we further show that this regime leads to variance advantages over direct human evaluation.

Figure 3 illustrates how the confidence-interval length decreases as the calibration dataset grows. The dashed curves correspond to calibration datasets with equal numbers of ‘incorrect’ and ‘correct’ instances ($m_0 = m_1$), using $\hat{q}_0 = 0.7$, $\hat{q}_1 = 0.9$, and test-set accuracies $\hat{p} \in \{0.3, 0.5, 0.7, 0.9\}$. For example, the red dashed curve ($\hat{p} = 0.3$) indicates that achieving an interval shorter than 0.1 requires approximately $m = m_0 + m_1 \approx 200$ calibration examples. Because the calibration dataset is collected independently and its label composition can sometimes be influenced through sampling strategies, asymmetric label sizes ($m_0 \neq m_1$) are feasible. This flexibility is important, as the two label types typically contribute asymmetrically to the overall uncertainty.

Adaptive Allocation to Reduce Confidence-Interval Length. Motivated by this observation, we introduce an adaptive allocation procedure in Algorithm 1. The algorithm first collects a small pilot calibration sample (e.g., $m_{\text{pilot}} = 10$ per label type) to obtain preliminary estimates $(\tilde{q}_0, \tilde{q}_1)$, from which it computes the empirical error ratio $(1 - \tilde{q}_0)/(1 - \tilde{q}_1)$. Combining this ratio with the naive accuracy \hat{p} from the test set, the algorithm estimates the optimal allocation of (m_0, m_1) that approximately minimizes the confidence-interval length in (6). As shown by the solid curves in Figure 3, this adaptive allocation yields shorter intervals under a fixed calibration budget than symmetric allocation. The optimality of this rule is established in the following section.

A Python implementation that computes the bias-adjusted estimator $\hat{\theta}$ in (5) and the confidence interval in (6) is provided in Appendix C.

5. Theoretical Justification of Our Method

We provide theoretical guarantees for the bias-adjusted estimator and confidence interval introduced in Section 4. We analyze bias properties, derive asymptotic variance and confidence intervals, and establish the optimality of the proposed label allocation rule for the calibration dataset.

5.1. Mitigating Bias on Point Estimator.

We compare the bias-adjusted estimator $\hat{\theta}$ in (5) with the naive estimator \hat{p} in (2). By the law of total probability,

$$\begin{aligned} p &= \Pr(\hat{Z} = 1 | Z = 1) \cdot \Pr(Z = 1) \\ &\quad + (1 - \Pr(\hat{Z} = 0 | Z = 0)) \cdot (1 - \Pr(Z = 1)) \\ &= (q_0 + q_1 - 1) \cdot \theta + (1 - q_0). \end{aligned}$$

Thus, $\mathbb{E}[\hat{p}] = \theta$ for all θ if and only if the LLM judge is perfect, i.e., $q_0 = q_1 = 1$; otherwise, \hat{p} is biased. When $q_0 + q_1 < 2$, the expectation can be rewritten as

$$\mathbb{E}[\hat{p}] = p = \theta + (2 - q_0 - q_1) \left(\frac{1 - q_0}{2 - q_0 - q_1} - \theta \right),$$

which makes the direction of the bias explicit: when the true accuracy θ is smaller than the threshold $\frac{1 - q_0}{2 - q_0 - q_1}$, the estimator \hat{p} exhibits a positive bias, i.e., $\mathbb{E}[\hat{p}] > \theta$; conversely, when θ exceeds this threshold, the bias becomes negative.

Bias-adjusted Estimator. Assuming $q_0 + q_1 > 1$, inverting the above relation gives the estimator in Eq. (4). Replacing (p, q_0, q_1) with their empirical estimates gives the bias-adjusted estimator $\hat{\theta}$ in (5) (Rogan & Gladen, 1978; Lang & Reiczigel, 2014). When q_0 and q_1 are known, $\hat{\theta}$ is unbiased; when they are estimated from a calibration dataset, residual bias remains.

The following result shows that, even with estimated sensitivity and specificity, the adjusted estimator $\hat{\theta}$ achieves smaller bias than the naive estimator as the calibration dataset grows. All proofs are provided in Appendix B.

Proposition 5.1. *Suppose that $m := 2m_0 = 2m_1$ and that $q := q_0 = q_1$ with $0.5 < q \leq 1$. For sufficiently large $m \gtrsim 2q/(2q - 1)^2$, the absolute bias of $\hat{\theta}$ in (1) is always smaller than that of \hat{p} in (2) for all θ .*

Even when \hat{q}_0 and \hat{q}_1 are estimated from data, the adjusted estimator $\hat{\theta}$ exhibits lower bias than the naive estimator once the calibration dataset is large enough. Moreover, the required calibration size depends on judge reliability: fewer samples suffice when the LLM has high sensitivity and specificity ($q \approx 1$), while substantially more are required as performance approaches chance level ($q \approx 0.5$).

5.2. Uncertainty Quantification via Confidence Interval

We quantify uncertainty in $\hat{\theta}$ arising from two sources: the test dataset used to estimate p and the calibration dataset used to estimate q_0 and q_1 . Applying the delta method (Dorfman, 1938; Ver Hoef, 2012) gives the asymptotic variance

$$\text{Var}(\hat{\theta}) = \frac{\frac{\hat{p}(1-\hat{p})}{n} + (1 - \hat{\theta})^2 \cdot \frac{\hat{q}_0(1-\hat{q}_0)}{m_0} + \hat{\theta}^2 \cdot \frac{\hat{q}_1(1-\hat{q}_1)}{m_1}}{(\hat{q}_0 + \hat{q}_1 - 1)^2}, \quad (9)$$

using the binomial variance formulas for \hat{p} , \hat{q}_0 , and \hat{q}_1 . A detailed derivation is provided in Appendix B.1.

Based on this variance, we construct a confidence interval for θ using the “add two successes and two failures” adjusted Wald approach (de Laplace, 1820; Agresti & Coull, 1998; Brown et al., 2001; Lang & Reiczigel, 2014). Specifically, we replace \hat{p} , \hat{q}_0 , and \hat{q}_1 with their adjusted versions \tilde{p} , \tilde{q}_0 , and \tilde{q}_1 , as defined in Eq. (7). These adjustments can be interpreted as adding one (or $z_\alpha^2/2$) success and one (or $z_\alpha^2/2$) failure to each estimate, improving coverage accuracy for small sample sizes (Agresti & Caffo, 2000). Substituting the estimates gives the confidence interval in Eq. (6).

The adjustment also induces a small shift in the interval center (i.e., $d\tilde{\theta}$ in Eq. (8)) due to the dependence of $\hat{\theta}$ on \hat{q}_0 and \hat{q}_1 . Its effect on interval length is negligible and therefore ignored in the final approximation, see Lang & Reiczigel (2014) for details.

The Optimal Allocation of the Calibration Dataset. We characterize how to allocate calibration label sizes under a fixed budget to minimize the confidence-interval length. Assume that \tilde{q}_0, \tilde{q}_1 are close to one and define the error ratio $\kappa := (1 - \tilde{q}_0)/(1 - \tilde{q}_1)$.

Proposition 5.2. *Suppose that \tilde{q}_0 and \tilde{q}_1 are close to 1. Then the minimum length of the confidence interval defined in (6) is achieved when $\tilde{m}_0 \approx (1/\tilde{p} - 1)\sqrt{\kappa} \cdot \tilde{m}_1$.*

This result provides a guideline for allocating calibration samples between the two response types. When the LLM is less accurate at identifying ‘incorrect’ responses (i.e., when the error ratio κ is large), more calibration samples should be allocated to estimating \tilde{q}_0 . Moreover, when the proportion \tilde{p} of ‘correct’ predictions in the test set is small, the factor $(1/\tilde{p} - 1)$ amplifies the contribution of \tilde{m}_0 to the interval length, again favoring additional calibration samples for ‘incorrect’ responses.

To implement this rule in practice, we use the adaptive procedure in Algorithm 1. The procedure estimates $(\tilde{q}_0, \tilde{q}_1)$ from a small pilot calibration sample, computes the error ratio $\hat{\kappa}$, and allocates (m_0, m_1) according to the approximate optimal ratio implied by Proposition 5.2.

6. When Are LLM-as-a-Judge Evaluations Preferable to Fully Human Evaluation?

In Section 5, we show that a human-evaluated calibration dataset is essential for reliable LLM-as-a-judge evaluation, as it enables correction of bias arising from judgment errors. However, this requirement raises a natural question. If human annotators are already available to label a set of instances, one could instead directly estimate the target accuracy θ by evaluating responses using human labels.

More concretely, given a fixed evaluation budget that allows labeling m responses, there are two possible strategies: (i) use the m labeled instances as a calibration dataset to estimate the sensitivity and specificity of an LLM judge and then correct large-scale LLM-based evaluations, or (ii) directly evaluate m test instances with human labels and estimate the target accuracy from these samples alone.

Since a calibration dataset is required for reliable LLM-as-a-judge evaluation, determining which of these two strategies is preferable is a fundamental question. We address this question from a statistical perspective and show that there exist parameter regimes in which, for the same evaluation budget m , LLM-as-a-judge evaluation with calibration is statistically more efficient than fully human evaluation, yielding more precise accuracy estimates.

Regimes in Which LLM-as-a-Judge Evaluations Achieve Lower Variance in Estimation.

Any estimator based on a finite sample incurs non-negligible variance. In fully human evaluation, this variance cannot be arbitrarily reduced because human annotation is costly and does not scale.

In contrast, in LLM-as-a-judge evaluations, the test dataset can be scaled at negligible cost. As the test-set size increases, the variance contributed by LLM-based judgments becomes negligible, and the dominant source of uncertainty arises from bias correction using a finite human-labeled calibration dataset. This variance decreases as the LLM judge becomes more reliable, i.e., as its sensitivity q_1 and specificity q_0 approach one. Aggregating large-scale LLM judgments with bias correction can yield an estimator with *lower variance than one based solely on human evaluation*.

Figure 4 visualizes this comparison. The shaded region indicates values of the true accuracy θ for which the LLM-as-a-judge estimator under our framework achieves lower variance than direct human evaluation. This region is centered around $\theta = 1/2$, where evaluation is intrinsically most uncertain, and expands as the LLM judge becomes more accurate. Intuitively, when the correctness of a test instance is ambiguous and the LLM judge is reliable, aggregating many inexpensive LLM judgments with bias correction is more statistically efficient than human annotation alone.

The following proposition formalizes this comparison by

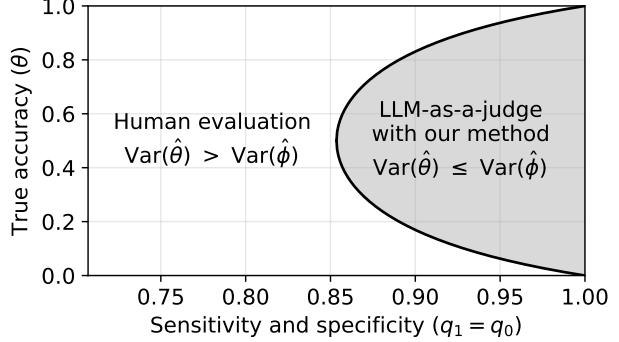


Figure 4. Comparison of estimator variances between LLM-as-a-judge evaluation under our method ($\hat{\theta}$ in Eq. (7)) and direct human evaluation ($\hat{\phi}$ in Proposition 6.1). The shaded regions indicate parameter regimes where the LLM-as-a-judge estimator under our method has lower variance than direct human evaluation.

characterizing the parameter regimes in which LLM-as-a-judge evaluation with calibration is statistically more efficient than a fully human evaluation.

Proposition 6.1. Suppose that $n \rightarrow \infty$ and that $q := q_0 = q_1$ with $\frac{1}{2} + \frac{1}{2\sqrt{2}} < q \leq 1$. Let $M_1 \sim \text{Binomial}(m, \theta)$ denote the number of correct instances in a calibration dataset of size m , and define $\hat{\phi} := M_1/m$. Let $\hat{\theta}$ be the bias-adjusted estimator in Eq. (7). Then, for sufficiently large m , $\text{Var}(\hat{\theta}) \leq \text{Var}(\hat{\phi})$ if and only if

$$\theta \in \left[\frac{1}{2} - \sqrt{\frac{1}{2} - \frac{1}{4(2q-1)^2}}, \frac{1}{2} + \sqrt{\frac{1}{2} - \frac{1}{4(2q-1)^2}} \right].$$

7. Robust Estimation under Distribution Shift

Having established when LLM-as-a-judge evaluation is preferable to human evaluation, we now consider which LLM-as-a-judge estimators should be preferred. To this end, we compare our estimator with existing calibration-based approaches, highlighting differences in the distributional assumptions required for unbiasedness and how these assumptions behave under distribution shift between the test distribution \mathbb{P} and the calibration distribution \mathbb{Q} .

Data-Generating Process for LLM-as-a-Judge Evaluation. LLM-as-a-judge evaluation can be formalized through a data-generating process consisting of two probabilistic components: the marginal distribution of true labels $\Pr(Z)$, which is determined by the dataset, and the conditional behavior of the LLM judge, $\Pr(\hat{Z} | Z, \xi)$. Here, ξ denotes auxiliary factors that may influence judgments, such as response length or stylistic attributes (Zhou et al., 2023; Dubois et al., 2024).

Throughout the main analysis, we *only* assume that the judge’s behavior depends only on the true label and is invariant across datasets:

$$\Pr_{\mathbb{P}}(\hat{Z} | Z) = \Pr_{\mathbb{Q}}(\hat{Z} | Z).$$

Table 1. Comparison of point estimators for estimating $\mathbb{E}_{\mathbb{P}}[Z]$, together with their estimation formulas and assumptions for unbiasedness. Here, \mathbb{P} denotes the test distribution and \mathbb{Q} the calibration distribution. Estimators (ii)–(iv) require $\Pr_{\mathbb{P}}(Z) = \Pr_{\mathbb{Q}}(Z)$ for their assumptions to hold, whereas the misclassification-adjusted estimator in (v), used in our method, remains valid when $\Pr_{\mathbb{P}}(Z) \neq \Pr_{\mathbb{Q}}(Z)$.

Name	Estimation formula	Assumption
(i) Naive LLM judgment estimator	$\mathbb{E}_{\mathbb{P}}[\hat{Z}]$	$\Pr_{\mathbb{P}}(Z) = \Pr_{\mathbb{P}}(\hat{Z})$
(ii) Calibration-only estimator	$\mathbb{E}_{\mathbb{Q}}[Z]$	$\Pr_{\mathbb{P}}(Z) = \Pr_{\mathbb{Q}}(Z)$
(iii) Difference estimator (Angelopoulos et al., 2023a)	$\mathbb{E}_{\mathbb{P}}[\hat{Z}] + \mathbb{E}_{\mathbb{Q}}[Z - \hat{Z}]$	$\Pr_{\mathbb{P}}(Z - \hat{Z}) = \Pr_{\mathbb{Q}}(Z - \hat{Z})$
(iv) Conditional calibration estimator	$\Pr_{\mathbb{Q}}(Z \hat{Z}) \mathbb{E}_{\mathbb{P}}[\hat{Z}]$	$\Pr_{\mathbb{P}}(Z \hat{Z}) = \Pr_{\mathbb{Q}}(Z \hat{Z})$
(v) Misclassification-adjusted estimator (Ours; extending Rogan & Gladen, 1978)	$(\Pr_{\mathbb{Q}}(\hat{Z} Z))^{-1} \mathbb{E}_{\mathbb{P}}[\hat{Z}]$	$\Pr_{\mathbb{P}}(\hat{Z} Z) = \Pr_{\mathbb{Q}}(\hat{Z} Z)$

This assumption reflects the common practice of using the same LLM judge and prompting strategy for both test and calibration datasets. We discuss a relaxation of this assumption that allows $\Pr(\hat{Z} | Z, \xi)$ to depend on ξ in Section 9.

In contrast, we allow the label distributions to differ:

$$\Pr_{\mathbb{P}}(Z) \neq \Pr_{\mathbb{Q}}(Z). \quad (10)$$

Such distribution shift can arise because calibration datasets are curated to facilitate accurate error estimation (e.g., through class balancing or simplified examples), whereas test datasets are designed to reflect realistic evaluation scenarios (Jung et al., 2024).

Under this setup, Bayes’ rule implies

$$\Pr(Z | \hat{Z}) \propto \Pr(\hat{Z} | Z) \Pr(Z),$$

showing that the posterior distribution depends on the label distribution $\Pr(Z)$. Consequently, even if $\Pr(\hat{Z} | Z)$ is stable across datasets, a shift in $\Pr(Z)$ generally induces a shift in the posterior, i.e., $\Pr_{\mathbb{P}}(Z | \hat{Z}) \neq \Pr_{\mathbb{Q}}(Z | \hat{Z})$. Estimators that assume invariance of $\Pr(Z | \hat{Z})$ are therefore sensitive to distribution shift described in Eq. (10).

Implicit Assumptions of Existing Estimators. We compare several point estimators for estimating $\mathbb{E}_{\mathbb{P}}[Z]$ under the test distribution, as studied in prior work (Kloos et al., 2021; Meertens et al., 2022). To facilitate comparison, define

$$\mathbf{Z} := (Z, 1 - Z)^\top, \quad \hat{\mathbf{Z}} := (\hat{Z}, 1 - \hat{Z})^\top,$$

and let $\Pr(\hat{\mathbf{Z}} | \mathbf{Z})$ and $\Pr(\mathbf{Z} | \hat{\mathbf{Z}})$ denote the confusion and calibration matrices, respectively. In particular, the (a, b) -th entry of the confusion matrix is $[\Pr(\hat{\mathbf{Z}} | \mathbf{Z})]_{a,b} := \Pr([\hat{\mathbf{Z}}]_a | [\mathbf{Z}]_b)$, where $[\hat{\mathbf{Z}}]_a$ denotes the a -th component of $\hat{\mathbf{Z}}$.

Table 1 summarizes these estimators, along with their formulas and the invariance assumptions required for unbiasedness. As discussed earlier, the naive estimator in (i), $\hat{p} = [\mathbb{E}_{\mathbb{P}}[\hat{Z}]]_1$, is biased when the LLM judge is imperfect.

Effect of Distribution Shift on Estimator Bias. Most existing calibration-based estimators rely on invariance assumptions that fail under the distribution shift described in

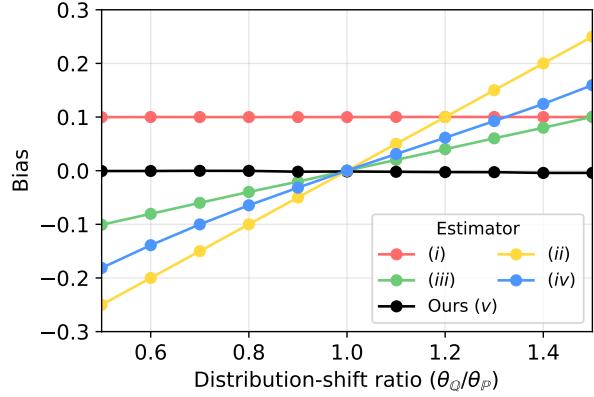


Figure 5. Effects of distribution shift $\Pr_{\mathbb{P}}(Z) \neq \Pr_{\mathbb{Q}}(Z)$ on estimator bias. We fix the true test-set accuracy at $\theta_{\mathbb{P}} = 0.5$ and vary the calibration-set accuracy $\theta_{\mathbb{Q}} \in [0.25, 0.75]$, corresponding to a distribution-shift ratio from 0.5 to 1.5. The misclassification-adjusted estimator in (v), used in our method, remains unbiased.

Eq. (10), even if the judge’s conditional behavior remains stable. In particular, the calibration-only estimator in (ii), the difference estimator in (iii), and the conditional calibration estimator in (iv) all require invariance of the marginal label distribution $\Pr(Z)$. This assumption is violated under the distribution shift.

In contrast, the misclassification-adjusted estimator in (v), used in our method, requires only stability of the confusion matrix $\Pr(\hat{\mathbf{Z}} | \mathbf{Z})$ across datasets. Because it does not depend on the label distribution $\Pr(Z)$, it remains unbiased even when $\Pr_{\mathbb{P}}(Z) \neq \Pr_{\mathbb{Q}}(Z)$. In Appendix B.3, we show that this estimator, used in our method, is equivalent to our point estimator defined in Eq. (5).

To illustrate this robustness, consider a setting in which the test and calibration datasets share the same conditional judge behavior with $(q_0, q_1) = (0.7, 0.9)$, but differ in their true accuracies: $\theta_{\mathbb{P}} = 0.5$ for the test dataset and $\theta_{\mathbb{Q}} \in [0.25, 0.75]$ for the calibration dataset. The resulting biases are shown in Figure 5. Under distribution shift, the calibration-only estimator (ii), difference estimator (iii) and the difference estimator (iv) exhibit bias, whereas the misclassification-adjusted estimator (v) remains unbiased, as it does not depend on the marginal distribution of Z . In other words, it remains valid even when $\theta_{\mathbb{P}} \neq \theta_{\mathbb{Q}}$. Details of the simulation are provided in Appendix D.

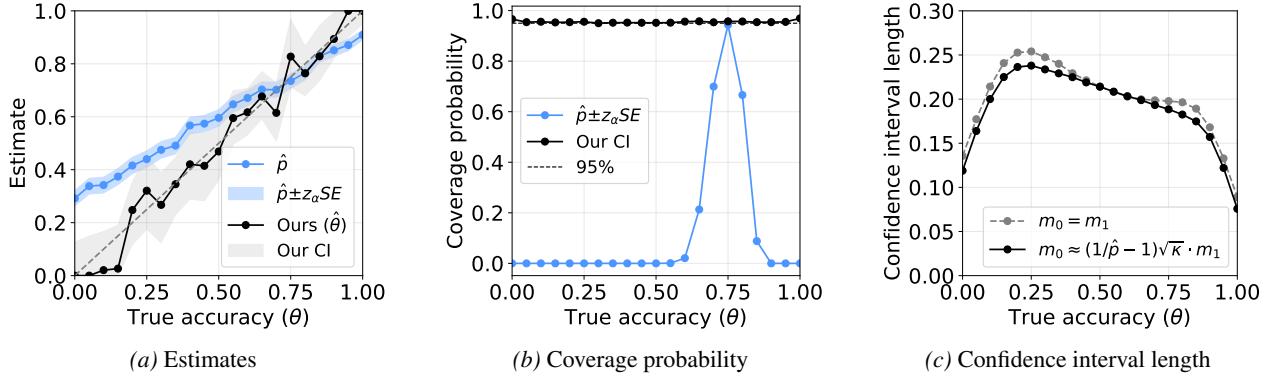


Figure 6. Monte Carlo simulation for estimating θ under an imperfect LLM judge with $(q_0, q_1) = (0.7, 0.9)$. We evaluate estimators across 21 values of $\theta \in [0, 1]$, each visualized as a single point. Figure 6a reports the results from a single run, while Figure 6b and Figure 6c summarize averages computed over 10,000 replications. All experiments use a test dataset of size $n = 1000$ and a calibration dataset of size $m = 200$, and we use an equal allocation $m_0 = m_1$ for Figure 6a and Figure 6c. (a) The naive estimator \hat{p} in Eq. (2) exhibits bias, while the unbiased estimator $\hat{\theta}$ in Eq. (5) closely recovers the true accuracy θ across all values. Shaded regions represent the 95% confidence intervals (CI). (b) Across all θ , the coverage probability of the confidence interval remains consistently close to the nominal 95% level. (c) Given a fixed calibration budget of $m = 200$, we compare two allocation strategies: an equal split ($m_0 = m_1$) and the allocation proportional to $m_0 \propto (1/\hat{p} - 1)\sqrt{\kappa} \cdot m_1$ by using Algorithm 1. The proposed allocation gives shorter confidence intervals.

8. Empirical Validation on Our Method

To validate the theoretical results established in Section 5, we empirically evaluate the proposed estimator, confidence interval, and calibration allocation strategy through Monte Carlo simulation and real-world benchmarks.

8.1. Monte Carlo Simulation

Experimental Setup. We evaluate the proposed method under the following parameter configuration. The LLM judge is characterized by parameters $(q_0, q_1) = (0.7, 0.9)$, and the true accuracy varies over $\theta \in \{0, 0.05, 0.10, \dots, 1\}$, resulting in 21 distinct settings. For each configuration of (q_0, q_1, θ) , we generate a test dataset of size $n = 1000$ and a calibration dataset of total size $m = 200$, with equal allocation $m_0 = m_1$ unless stated otherwise.

We compute the naive estimator \hat{p} in (2) together with its confidence interval, and compute the bias-adjusted estimator $\hat{\theta}$ in (5) and the confidence interval in (6). Each configuration is replicated 10,000 times to evaluate estimates, interval coverage, and interval length. Additional simulations under alternative settings are provided in Appendix E.

Bias Reduction in Point Estimation. Figure 6a compares the naive estimator \hat{p} and the bias-adjusted estimator $\hat{\theta}$ based on a single simulation run. As shown in Section 5, \hat{p} exhibits bias, particularly overestimating the true accuracy when the underlying θ is small. In contrast, $\hat{\theta}$ closely aligns with the true accuracy across all values of θ , demonstrating the bias correction achieved by (5).

Coverage of the Confidence Interval. Figure 6b reports the empirical coverage probability of the confidence interval in (6). Across all values of θ , the coverage remains

consistently close to the nominal 95% level, whereas the confidence interval constructed from the naive estimator \hat{p} achieves nearly zero coverage except at a few values of θ . These results confirm that the proposed confidence interval provides reliable uncertainty quantification.

Efficiency of Optimal Calibration Allocation. To examine the benefits of optimal calibration allocation, we compare two strategies under a fixed calibration budget of $m = 200$: (i) an equal allocation ($m_0 = m_1 = 100$), and (ii) the allocation produced by Algorithm 1, which approximates the optimal ratio derived in Proposition 5.2. Figure 6c shows that the adaptive allocation consistently gives shorter confidence intervals than the equal-split baseline.

8.2. Chatbot Arena Benchmark

We evaluate our method on the Chatbot Arena benchmark (Chiang et al., 2024), a crowdsourced platform widely used in LLM-as-a-judge studies (Zheng et al., 2023). In Chatbot Arena, users vote between responses generated by two anonymous models to the same prompt. These pairwise comparisons are aggregated to estimate each model’s win rate, defined as the probability that the target model’s response is preferred over its opponent’s.

Experimental Setup. We conduct experiments on six models used in Zheng et al. (2023): Alpaca-13B (Taori et al., 2023), Claude-v1, FastChat-T5-3B, GPT-4, LLaMA-13B (Touvron et al., 2023), and Vicuna-13B (Chiang et al., 2023). For each target model, the dataset consists of response pairs composed of one response generated by the target model and one generated by a randomly selected opponent model.

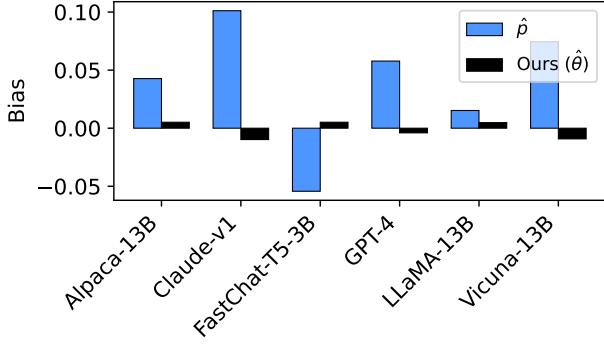


Figure 7. Average bias of winning rates for six models on Chatbot Arena, averaged over 100 random test (90%) / calibration (10%) splits. We compare the raw LLM judge estimator \hat{p} and the proposed bias-adjusted estimator $\hat{\theta}$, using GPT-4.1-mini as the LLM judge. The proposed method reduces bias across all models.

Table 2. 95% confidence interval coverage and average CI length of winning rate estimates on Chatbot Arena, averaged over 100 random replications.

Model	Alpaca	Claude	FastChat-T5	GPT-4	LLaMA	Vicuna
Coverage	96%	95%	97%	97%	99%	99%
CI length	0.193	0.262	0.256	0.210	0.267	0.178

To quantify bias, we consider question-response pairs with human preference evaluations, which serve as the ground truth for bias computation. Each response pair is additionally evaluated by GPT-4.1-mini, which acts as the LLM judge whose bias we aim to analyze and correct.

The dataset is randomly split into a test set (90%) and a calibration set (10%). The calibration set is used to estimate parameters for bias adjustment, while bias is evaluated on the test set. We repeat this process 100 times with different random splits and report averaged results.

We compare two estimators: the baseline estimator \hat{p} , which directly uses the LLM judge’s preference probabilities, and our $\hat{\theta}$ which applies the proposed bias-adjustment method.

Results. Figure 7 shows the average bias of winning rate for each model on the Chatbot Arena benchmark. Using raw judge scores \hat{p} , all models exhibit non-trivial bias with varying magnitude and direction. In contrast, the proposed bias-adjustment method $\hat{\theta}$ reduces bias across all six models. The improvement is consistent, indicating effective mitigation of judge-induced bias.

Table 2 reports the empirical coverage rates of the proposed 95% confidence intervals. Across all models, the coverage rates are close to or slightly above the nominal level, indicating that the proposed intervals preserve the desired significance level.

Overall, our method provides a simple and practical solution for bias correction in LLM-based evaluation. It operates directly on existing judge scores and requires only a small calibration dataset, enabling a principled statistical treatment of LLM-as-a-judge evaluation pipelines.

9. Conclusion

In LLM-as-a-judge evaluation, noisy judgments induce bias in naive point estimates. We introduce a bias-adjusted estimator that corrects for imperfect judgments and constructs confidence intervals that account for uncertainty arising in both the evaluation and calibration datasets. To reduce uncertainty, we show that focusing on the calibration design, such as the allocation between response types, is preferable to simply increasing the total number of judgments.

Beyond bias correction, we demonstrate that when the LLM judge is sufficiently accurate, our framework can achieve lower variance than direct human evaluation, and that the resulting estimator remains robust to distribution shift between the test and calibration datasets under a mild and realistic assumption on the LLM judge. We hope this work contributes to more reliable and transparent reporting practices in LLM-based evaluation.

Future Work. Several directions remain for future work. **(1)** Our method can be extended to account for auxiliary factors that influence LLM-as-a-judge evaluations beyond the true label Z . While our current analysis assumes that the LLM evaluation \hat{Z} depends only on the true label Z , in practice it may also be affected by additional nuisance factors ξ , such as response length or other stylistic attributes. This more realistic setting can be accommodated by allowing both the confusion matrix $\text{Pr}_{\mathbb{Q}}(\hat{Z} | Z, \xi)$ and the LLM evaluation $\mathbb{E}_{\mathbb{P}}[\hat{Z} | \xi]$ to depend on such factors. For example, one may estimate separate confusion matrices across different strata of ξ (e.g., short vs. long responses) and perform bias adjustment within each group, followed by aggregation across strata. Formally, this corresponds to correcting $\mathbb{E}_{\mathbb{P}}[\hat{Z} | \xi]$ using $\text{Pr}_{\mathbb{Q}}(\hat{Z} | Z, \xi)$ at the stratum level and averaging over ξ . **(2)** The proposed method can be extended to a multinomial setting by generalizing $(\text{Pr}_{\mathbb{Q}}(\hat{Z} | Z))^{-1}\mathbb{E}_{\mathbb{P}}[\hat{Z}]$ in Sec. 7 to a multivariate formulation, where Z and \hat{Z} are represented as probability distributions over multiple response categories. Such an extension would increase the number of parameters, and additional structural assumptions, such as constraints on the confusion matrix $\text{Pr}_{\mathbb{Q}}(\hat{Z} | Z)$, may be required to ensure stable estimation. **(3)** A conformal prediction framework could be incorporated to provide sample-specific uncertainty quantification by constructing statistically valid prediction regions around each bias-adjusted point estimate.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none of which we feel must be specifically highlighted here.

References

- Agresti, A. and Coull, B. Simple and effective confidence intervals for proportions and differences of proportions result from adding two successes and two failures. *The American Statistician*, 54(4):280–288, 2000.
- Agresti, A. and Coull, B. A. Approximate is better than “exact” for interval estimation of binomial proportions. *The American Statistician*, 52(2):119–126, 1998.
- Albinet, F. Why llms can actually judge other llms (and it’s not cheating), 8 2025. URL <https://franck-albi-net.pla.sh/post/llm-as-a-judge>.
- Angelopoulos, A. N., Bates, S., Fannjiang, C., Jordan, M. I., and Zrnic, T. Prediction-powered inference. *Science*, 382 (6671):669–674, 2023a.
- Angelopoulos, A. N., Duchi, J. C., and Zrnic, T. Ppi++: Efficient prediction-powered inference. *arXiv preprint arXiv:2311.01453*, 2023b.
- Boyeau, P., Angelopoulos, A. N., Li, T., Yosef, N., Malik, J., and Jordan, M. I. Autoeval done right: Using synthetic data for model evaluation. In *International Conference on Machine Learning*, 2025.
- Broska, D., Howes, M., and van Loon, A. The mixed subjects design: Treating large language models as potentially informative observations. *Sociological Methods & Research*, pp. 00491241251326865, 2025.
- Bross, I. Misclassification in 2 x 2 tables. *Biometrics*, 10 (4):478–486, 1954.
- Brown, L. D., Cai, T. T., and DasGupta, A. Interval Estimation for a Binomial Proportion. *Statistical Science*, 16 (2):101 – 133, 2001. doi: 10.1214/ss/1009213286. URL <https://doi.org/10.1214/ss/1009213286>.
- Buonaccorsi, J. P. *Measurement error: models, methods, and applications*. Chapman and Hall/CRC, 2010.
- Chiang, W.-L., Li, Z., Lin, Z., Sheng, Y., Wu, Z., Zhang, H., Zheng, L., Zhuang, S., Zhuang, Y., Gonzalez, J. E., Stoica, I., and Xing, E. P. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, March 2023. URL <https://lmsys.org/blog/2023-03-30-vicuna/>.
- Chiang, W.-L., Zheng, L., Sheng, Y., Angelopoulos, A. N., Li, T., Li, D., Zhu, B., Zhang, H., Jordan, M., Gonzalez, J. E., and Stoica, I. Chatbot arena: An open platform for evaluating LLMs by human preference. In *Forty-first International Conference on Machine Learning*, 2024. URL <https://openreview.net/forum?id=3MW8GKNyzI>.
- de Laplace, P. S. *Théorie analytique des probabilités*, volume 7. Courcier, 1820.
- Dorfman, R. A note on the δ -method for finding variance formulae. *Biometric Bulletin*, 1938.
- Dubois, Y., Liang, P., and Hashimoto, T. Length-controlled alpacaeval: A simple debiasing of automatic evaluators. In *First Conference on Language Modeling*, 2024. URL <https://openreview.net/forum?id=CybBmzWBX0>.
- Feng, Y., Wang, S., Cheng, Z., Wan, Y., and Chen, D. Are we on the right way to assessing llm-as-a-judge? *arXiv preprint arXiv:2512.16041*, 2025.
- Forman, G. Counting positives accurately despite inaccurate classification. In *European conference on machine learning*, pp. 564–575. Springer, 2005.
- Forman, G. Quantifying counts and costs via classification. *Data Mining and Knowledge Discovery*, 17(2):164–206, 2008.
- Fraser, C. Estimating how many there are of something when you can’t see them all perfectly, 11 2024. URL <https://colin-fraser.net/>.
- Godbole, A. and Jia, R. Verify with caution: The pitfalls of relying on imperfect factuality metrics. In Che, W., Nabende, J., Shutova, E., and Pilehvar, M. T. (eds.), *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 22889–22912, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-256-5. doi: 10.18653/v1/2025.findings-acl.1175. URL <https://aclanthology.org/2025.findings-acl.1175>.
- Gu, J., Jiang, X., Shi, Z., Tan, H., Zhai, X., Xu, C., Li, W., Shen, Y., Ma, S., Liu, H., Wang, S., Zhang, K., Wang, Y., Gao, W., Ni, L., and Guo, J. A survey on llm-as-a-judge, 2025. URL <https://arxiv.org/abs/2411.15594>.
- Huang, H., Bu, X., Zhou, H., Qu, Y., Liu, J., Yang, M., Xu, B., and Zhao, T. An empirical study of LLM-as-a-judge for LLM evaluation: Fine-tuned judge model is not a general substitute for GPT-4. In *Findings of the Association for Computational Linguistics*, July 2025. ISBN 979-8-89176-256-5. doi: 10.18653/v1/2025.findings-acl.

306. URL <https://aclanthology.org/2025.findings-acl.306/>.
- Jung, J., Brahman, F., and Choi, Y. Trust or escalate: Llm judges with provable guarantees for human agreement. *arXiv preprint arXiv:2407.18370*, 2024.
- Kloos, K., Meertens, Q., Scholtus, S., and Karch, J. Comparing correction methods to reduce misclassification bias. In *Artificial Intelligence and Machine Learning*, pp. 64–90, Cham, 2021. Springer International Publishing.
- Koo, R., Lee, M., Raheja, V., Park, J. I., Kim, Z. M., and Kang, D. Benchmarking cognitive biases in large language models as evaluators. In *Findings of the Association for Computational Linguistics*, August 2024. doi: 10.18653/v1/2024.findings-acl.29. URL <https://aclanthology.org/2024.findings-acl.29/>.
- Lang, Z. and Reiczigel, J. Confidence limits for prevalence of disease adjusted for estimated sensitivity and specificity. *Preventive Veterinary Medicine*, 113(1):13–22, 2014. ISSN 0167-5877. doi: <https://doi.org/10.1016/j.prevetmed.2013.09.015>. URL <https://www.sciencedirect.com/science/article/pii/S0167587713002936>.
- Li, D., Jiang, B., Huang, L., Beigi, A., Zhao, C., Tan, Z., Bhattacharjee, A., Jiang, Y., Chen, C., Wu, T., Shu, K., Cheng, L., and Liu, H. From generation to judgment: Opportunities and challenges of LLM-as-a-judge. In *Empirical Methods in Natural Language Processing*, November 2025. doi: 10.18653/v1/2025.emnlp-main.138. URL <https://aclanthology.org/2025.emnlp-main.138/>.
- Liu, Y., Iter, D., Xu, Y., Wang, S., Xu, R., and Zhu, C. G-eval: NLG evaluation using gpt-4 with better human alignment. In *Empirical Methods in Natural Language Processing*, December 2023. doi: 10.18653/v1/2023.emnlp-main.153. URL <https://aclanthology.org/2023.emnlp-main.153/>.
- Meertens, Q., Diks, C., Van Den Herik, H., and Takes, F. Improving the output quality of official statistics based on machine learning algorithms. *Journal of Official Statistics*, 38(2):485–508, 2022.
- Miller, E. Adding error bars to evals: A statistical approach to language model evaluations. *arXiv preprint arXiv:2411.00640*, 2024.
- Mukherjee, A., Bullo, M., Basu, D., and Gündüz, D. Test-time verification via optimal transport: Coverage, roc, & sub-optimality. *arXiv preprint arXiv:2510.18982*, 2025.
- Rogan, W. J. and Gladen, B. Estimating prevalence from the results of a screening test. *American Journal of Epidemiology*, 107(1):71–76, 01 1978. ISSN 0002-9262. doi: 10.1093/oxfordjournals.aje.a112510. URL <https://doi.org/10.1093/oxfordjournals.aje.a112510>.
- Schwartz, J. E. The neglected problem of measurement error in categorical data. *Sociological Methods & Research*, 13(4):435–466, 1985.
- Taori, R., Gulrajani, I., Zhang, T., Dubois, Y., Li, X., Guestrin, C., Liang, P., and Hashimoto, T. B. Stanford alpaca: An instruction-following llama model. https://github.com/tatsu-lab/stanford_alpaca, 2023.
- Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.
- Ver Hoef, J. M. Who invented the delta method? *The American Statistician*, 66(2):124–127, 2012.
- Wang, J., Liang, Y., Meng, F., Sun, Z., Shi, H., Li, Z., Xu, J., Qu, J., and Zhou, J. Is ChatGPT a good NLG evaluator? a preliminary study. In *Proceedings of the 4th New Frontiers in Summarization Workshop*, December 2023. doi: 10.18653/v1/2023.newsum-1.1. URL <https://aclanthology.org/2023.newsum-1.1/>.
- Wang, P., Li, L., Chen, L., Cai, Z., Zhu, D., Lin, B., Cao, Y., Kong, L., Liu, Q., Liu, T., and Sui, Z. Large language models are not fair evaluators. In *Annual Meeting of the Association for Computational Linguistics*, August 2024. doi: 10.18653/v1/2024.acl-long.511. URL [https://aclanthology.org/2024.acl-long.511/](https://aclanthology.org/2024.acl-long.511).
- Zheng, L., Chiang, W.-L., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., Lin, Z., Li, Z., Li, D., Xing, E., Zhang, H., Gonzalez, J. E., and Stoica, I. Judging llm-as-a-judge with mt-bench and chatbot arena. In *Advances in Neural Information Processing Systems*, 2023.
- Zhou, J., Lu, T., Mishra, S., Brahma, S., Basu, S., Luan, Y., Zhou, D., and Hou, L. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*, 2023.
- Zrnic, T. and Candès, E. J. Cross-prediction-powered inference. *Proceedings of the National Academy of Sciences*, 121(15):e2322083121, 2024.

A. Adaptive Allocation of Calibration Samples

We propose an adaptive procedure for allocating calibration samples between the two ground-truth classes ('*correct*' and '*incorrect*') using a small pilot calibration set. Leveraging pilot estimates of sensitivity \tilde{q}_1 and specificity \tilde{q}_0 in Algorithm 1, the algorithm attains an optimal allocation under a fixed total calibration budget, leading to short confidence intervals.

Algorithm 1 Adaptive allocation of calibration samples

Input: Total calibration budget m , pilot sample size $2m_{\text{pilot}}$ with $2m_{\text{pilot}} \leq m$, and the estimate \hat{p} from the test dataset.

Output: Allocated calibration sample sizes (m_0, m_1) .

1: **Pilot calibration.**

2: Collect m_{pilot} calibration examples with true label $z_j = 0$, and m_{pilot} examples with $z_j = 1$.

3: Compute \tilde{q}_0 and \tilde{q}_1 :

$$\tilde{q}_0 = \frac{\sum_{z_j=0} \mathbf{1}\{\hat{z}_j = 0, z_j = 0\} + 1}{m_{\text{pilot}} + 2}, \quad \tilde{q}_1 = \frac{\sum_{z_j=1} \mathbf{1}\{\hat{z}_j = 1, z_j = 1\} + 1}{m_{\text{pilot}} + 2}.$$

4: Compute the estimated error ratio:

$$\hat{\kappa} = \frac{1 - \tilde{q}_0}{1 - \tilde{q}_1}.$$

5: **Compute adaptive allocation.**

6: Using the approximation in Proposition 5.2, compute the provisional allocation:

$$m_1^* = \text{round} \left(\frac{m}{1 + (1/\hat{p} - 1)\sqrt{\hat{\kappa}}} \right).$$

7: Enforce pilot size:

$$m_1 = \min \{ \max\{m_1^*, m_{\text{pilot}}\}, m - m_{\text{pilot}} \}, \quad m_0 = m - m_1.$$

B. Proofs

B.1. Deriving the Variance of Estimators

Because p follows a binomial distribution, the variance of \hat{p} is

$$\text{Var}(\hat{p}) = \hat{p}(1 - \hat{p})/n.$$

Similarly, we have $\text{Var}(\hat{q}_0) = \hat{q}_0(1 - \hat{q}_0)/m_0$ and $\text{Var}(\hat{q}_1) = \hat{q}_1(1 - \hat{q}_1)/m_1$.

We now derive the asymptotic variance of $\hat{\theta}$ using the delta method (Dorfman, 1938; Ver Hoef, 2012) for $\hat{\theta} = \frac{\hat{p} + \hat{q}_0 - 1}{\hat{q}_0 + \hat{q}_1 - 1}$. The first order derivatives with respect to \hat{p} , \hat{q}_0 , and \hat{q}_1 are

$$\frac{\partial \hat{\theta}}{\partial \hat{p}} = \frac{1}{\hat{q}_0 + \hat{q}_1 - 1}, \quad \frac{\partial \hat{\theta}}{\partial \hat{q}_0} = \frac{1 - \hat{\theta}}{\hat{q}_0 + \hat{q}_1 - 1}, \quad \frac{\partial \hat{\theta}}{\partial \hat{q}_1} = \frac{-\hat{\theta}}{\hat{q}_0 + \hat{q}_1 - 1}.$$

Assuming independence between the test dataset and the calibration dataset, the delta method gives

$$\text{Var}(\hat{\theta}) = \frac{\hat{p}(1 - \hat{p})/n + (1 - \hat{\theta})^2 \cdot \hat{q}_0(1 - \hat{q}_0)/m_0 + \hat{\theta}^2 \cdot \hat{q}_1(1 - \hat{q}_1)/m_1}{(\hat{q}_0 + \hat{q}_1 - 1)^2}.$$

B.2. Proofs of Propositions

Proposition B.1. Suppose that $m := 2m_0 = 2m_1$ and that $q := q_0 = q_1$ with $0.5 < q \leq 1$. For sufficiently large $m \gtrsim 2q/(2q-1)^2$, the absolute bias of $\hat{\theta}$ in (1) is always smaller than that of \hat{p} in (2) for all θ .

Proof. First, note that the bias of \hat{p} in (2) is

$$\mathbb{E}[\hat{p}] - \theta = (q_0 + q_1 - 1)\theta + (1 - q_0) - \theta = (2\theta - 1)(1 - q).$$

Next, consider the bias of $\hat{\theta}$ in (1). By the second-order delta method, we have

$$\begin{aligned}\mathbb{E}[\hat{\theta}] &\approx \frac{p + q_0 - 1}{q_0 + q_1 - 1} + \frac{1}{2} \left(-\frac{2(q_1 - p)}{(q_0 + q_1 - 1)^3} \cdot \frac{q_0(1 - q_0)}{m_0} + \frac{2(p + q_0 - 1)}{(q_0 + q_1 - 1)^3} \cdot \frac{q_1(1 - q_1)}{m_1} \right) \\ &= \theta - \frac{(q_1 - p)}{(q_0 + q_1 - 1)^3} \cdot \frac{q_0(1 - q_0)}{m_0} + \frac{(p + q_0 - 1)}{(q_0 + q_1 - 1)^3} \cdot \frac{q_1(1 - q_1)}{m_1},\end{aligned}$$

which implies

$$\mathbb{E}[\hat{\theta}] - \theta \approx \frac{-(1 - \theta)q_0(1 - q_0)/m_0 + \theta q_1(1 - q_1)/m_1}{(q_0 + q_1 - 1)^2} = \frac{1}{m} \cdot \frac{2q}{(2q-1)^2} \cdot (2\theta - 1)(1 - q).$$

Hence, for sufficiently large m satisfying $m \gtrsim 2q/(2q-1)^2$, we conclude the following for all θ :

$$|\mathbb{E}[\hat{\theta}] - \theta| \approx \left| \frac{1}{m} \cdot \frac{2q}{(2q-1)^2} \right| \cdot |(2\theta - 1)(1 - q)| < |(2\theta - 1)(1 - q)| = |\mathbb{E}[\hat{p}] - \theta|.$$

□

Proposition B.2. Suppose that \tilde{q}_0 and \tilde{q}_1 are close to 1, and let $\kappa := (1 - \tilde{q}_0)/(1 - \tilde{q}_1)$. Then the minimum length of the confidence interval defined in (6) is achieved when $\tilde{m}_0 \approx (1/\tilde{p} - 1)\sqrt{\kappa} \cdot \tilde{m}_1$.

Proof. The length of the confidence interval in (6) is given by

$$\begin{aligned}2z_\alpha \sqrt{\frac{\tilde{p}(1 - \tilde{p})/\tilde{n} + (1 - \tilde{\theta})^2 \cdot \tilde{q}_0(1 - \tilde{q}_0)/\tilde{m}_0 + \tilde{\theta}^2 \cdot \tilde{q}_1(1 - \tilde{q}_1)/\tilde{m}_1}{(\tilde{q}_0 + \tilde{q}_1 - 1)^2}} \\ \propto \sqrt{(1 - \tilde{\theta})^2 \cdot \tilde{q}_0(1 - \tilde{q}_0)/\tilde{m}_0 + \tilde{\theta}^2 \cdot \tilde{q}_1(1 - \tilde{q}_1)/\tilde{m}_1} \\ \propto \sqrt{(\tilde{q}_1 - \tilde{p})^2 \cdot \tilde{q}_0(1 - \tilde{q}_0)/\tilde{m}_0 + (\tilde{p} + \tilde{q}_0 - 1)^2 \cdot \tilde{q}_1(1 - \tilde{q}_1)/\tilde{m}_1}.\end{aligned}$$

By the arithmetic–geometric mean inequality, the minimum condition is satisfied when

$$\frac{\tilde{m}_0}{\tilde{m}_1} = \frac{|\tilde{q}_1 - \tilde{p}|}{|\tilde{p} + \tilde{q}_0 - 1|} \sqrt{\frac{\tilde{q}_0(1 - \tilde{q}_0)}{\tilde{q}_1(1 - \tilde{q}_1)}} \approx \frac{|1 - \tilde{p}|}{|\tilde{p}|} \sqrt{\frac{1 - \tilde{q}_0}{1 - \tilde{q}_1}} = (1/\tilde{p} - 1)\sqrt{\kappa},$$

where the approximation holds under the assumption that \tilde{q}_0 and \tilde{q}_1 are close to 1. □

Proposition B.3. Suppose that $n \rightarrow \infty$ and that $q := q_0 = q_1$ with $\frac{1}{2} + \frac{1}{2\sqrt{2}} < q \leq 1$. Let $M_1 \sim \text{Binomial}(m, \theta)$ denote the number of ‘correct’ instances in a calibration dataset, and define $\hat{\phi} := M_1/m$. Let $\hat{\theta}$ be the bias-adjusted estimator in Eq. (7). Then, for sufficiently large m , $\text{Var}(\hat{\theta}) \leq \text{Var}(\hat{\phi})$ if and only if

$$\theta \in \left[\frac{1}{2} - \sqrt{\frac{1}{2} - \frac{1}{4(2q-1)^2}}, \frac{1}{2} + \sqrt{\frac{1}{2} - \frac{1}{4(2q-1)^2}} \right].$$

Proof. Since $M_1 \sim \text{Binomial}(m, \theta)$, we have

$$\text{Var}(\hat{\phi}) = \frac{\theta(1-\theta)}{m}.$$

From Eq. (9), as $n \rightarrow \infty$ the contribution of \hat{p} vanishes. Then, we have

$$\text{Var}(\hat{\theta}) = \frac{(1-\theta)^2 \cdot \frac{q(1-q)}{m-M_1} + \theta^2 \cdot \frac{q(1-q)}{M_1}}{(2q-1)^2} \approx \frac{(1-\theta)^2 \cdot \frac{q(1-q)}{m(1-\theta)} + \theta^2 \cdot \frac{q(1-q)}{m\theta}}{(2q-1)^2} = \frac{1}{m} \cdot \frac{q(1-q)}{(2q-1)^2},$$

where the approximation follows from replacing M_1/m with its limit θ as $m \rightarrow \infty$.

Therefore, $\text{Var}(\hat{\theta}) \leq \text{Var}(\hat{\phi})$ is equivalent to

$$\theta(1-\theta) \geq \frac{q(1-q)}{(2q-1)^2}.$$

Solving this quadratic inequality gives

$$\theta \in \left[\frac{1}{2} - \sqrt{\frac{1}{2} - \frac{1}{4(2q-1)^2}}, \frac{1}{2} + \sqrt{\frac{1}{2} - \frac{1}{4(2q-1)^2}} \right].$$

Finally, note that the above interval is real-valued if and only if

$$q \geq \frac{1}{2} + \frac{1}{2\sqrt{2}}.$$

□

B.3. Equivalence between the Misclassification Estimator and Our Point Estimator

We show that the misclassification estimator $(\Pr_{\mathbb{Q}}(\hat{\mathbf{Z}} | \mathbf{Z}))^{-1} \mathbb{E}_{\mathbb{P}}[\hat{\mathbf{Z}}]$ in Section 7 is equivalent to our point estimator in Eq. (5). From the definition, we have

$$\begin{aligned} (\Pr_{\mathbb{Q}}(\hat{\mathbf{Z}} | \mathbf{Z}))^{-1} \mathbb{E}_{\mathbb{P}}[\hat{\mathbf{Z}}] &= \begin{pmatrix} \Pr_{\mathbb{Q}}(\hat{Z} = 1 | Z = 1) & \Pr_{\mathbb{Q}}(\hat{Z} = 1 | Z = 0) \\ \Pr_{\mathbb{Q}}(\hat{Z} = 0 | Z = 1) & \Pr_{\mathbb{Q}}(\hat{Z} = 0 | Z = 0) \end{pmatrix}^{-1} \begin{pmatrix} \Pr_{\mathbb{P}}(\hat{Z} = 1) \\ \Pr_{\mathbb{P}}(\hat{Z} = 0) \end{pmatrix} \\ &= \begin{pmatrix} \hat{q}_1 & 1 - \hat{q}_0 \\ 1 - \hat{q}_1 & \hat{q}_0 \end{pmatrix}^{-1} \begin{pmatrix} \hat{p} \\ 1 - \hat{p} \end{pmatrix} \\ &= \frac{1}{\hat{q}_0 + \hat{q}_1 - 1} \begin{pmatrix} \hat{q}_0 & -(1 - \hat{q}_0) \\ -(1 - \hat{q}_1) & \hat{q}_1 \end{pmatrix} \begin{pmatrix} \hat{p} \\ 1 - \hat{p} \end{pmatrix}, \end{aligned}$$

provided that $\hat{q}_0 + \hat{q}_1 \neq 1$. This gives

$$(\Pr_{\mathbb{Q}}(\hat{\mathbf{Z}} | \mathbf{Z}))^{-1} \mathbb{E}_{\mathbb{P}}[\hat{\mathbf{Z}}] = \frac{1}{\hat{q}_0 + \hat{q}_1 - 1} \begin{pmatrix} \hat{p} + \hat{q}_0 - 1 \\ \hat{q}_1 - \hat{p} \end{pmatrix}.$$

In particular, the first component corresponds to the estimator of $\theta = \Pr_{\mathbb{P}}(Z = 1)$:

$$\hat{\theta} = \frac{\hat{p} + \hat{q}_0 - 1}{\hat{q}_0 + \hat{q}_1 - 1},$$

which exactly matches our point estimator in Eq. (5).

C. Code

All code used for this paper, including a plug-in Python implementation of the introduced method for LLM-as-a-judge evaluation, is available in <https://github.com/UW-Madison-Lee-Lab/LLM-judge-reporting>. To make this appendix self-contained, we provide below the key functions that compute the bias-adjusted estimator and its confidence interval, corresponding to the method described in Section 4.

```

from math import sqrt
from scipy.stats import norm

def clip(x, low=0.0, high=1.0):
    return max(low, min(high, x))

def point_estimator(p, q0, q1):
    """Compute the adjusted point estimate."""
    th = (p+q0-1)/(q0+q1-1)
    return clip(th)

def confidence_interval(p, q0, q1, n, m0, m1, alpha=0.05):
    """Compute the adjusted (1 - alpha) confidence interval."""
    z = norm.ppf(1-alpha/2)
    p, q0, q1 = (n*p+z**2/2)/(n+z**2), (m0*q0+1)/(m0+2), (m1*q1+1)/(m1+2)
    n, m0, m1 = n+z**2, m0+2, m1+2
    th = (p+q0-1)/(q0+q1-1)
    dth = 2*z**2*(-(1-th)*q0*(1-q0)/m0+th*q1*(1-q1)/m1)
    se = sqrt((p*(1-p)/n+(1-th)**2*q0*(1-q0)/m0+th**2*q1*(1-q1)/m1)/(q0+q1-1))
    return clip(th+dth-z*se), clip(th-dth+z*se)

```

Figure 8. Python code implementation of the adjustment method described in Section 4 that computes the bias-adjusted estimate and the $(1 - \alpha)$ confidence interval for the true accuracy θ . The inputs p , q_0 , and q_1 are empirical estimates from the test and calibration datasets.

D. Experimental Setup for Distribution-Shift Analysis

Here, we describe the experimental setup used to produce the distribution-shift results in Figure 5. The experimental setup follows the Monte Carlo simulation in Section 8, with the calibration-set accuracy varied.

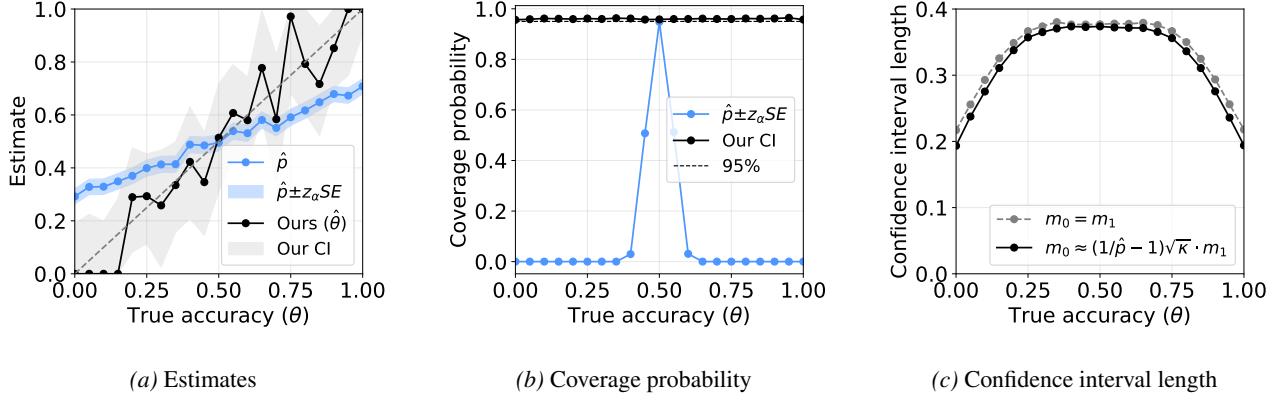
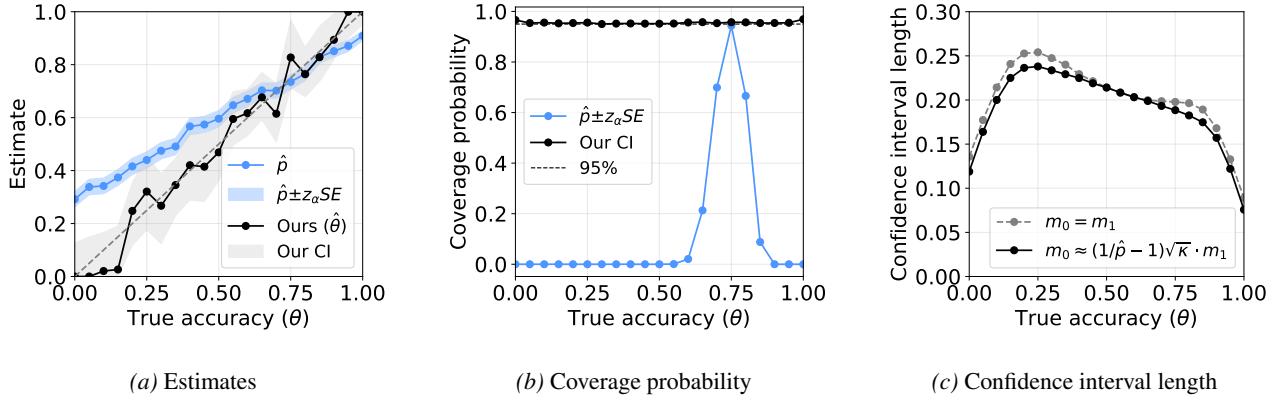
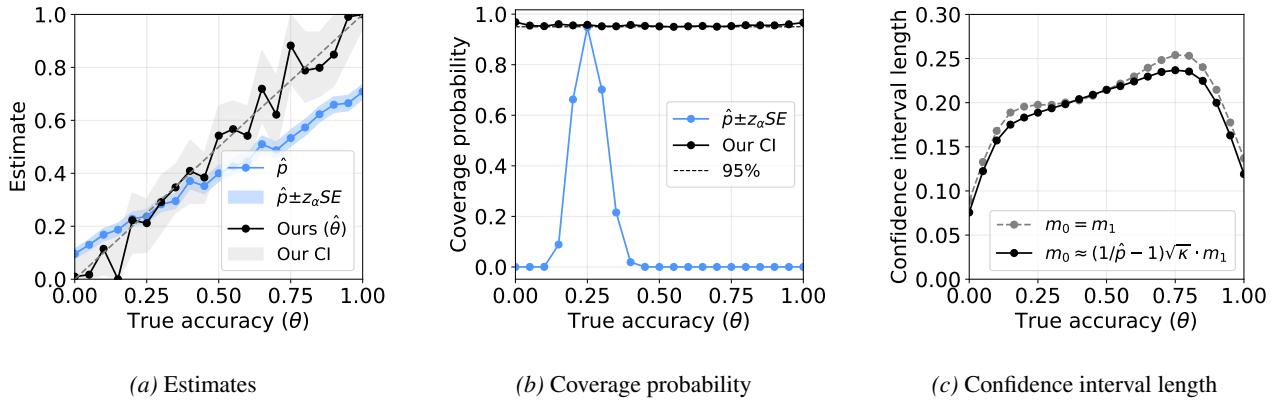
We fix the LLM judge behavior to $(q_0, q_1) = (0.7, 0.9)$ and the test-set accuracy to $\theta_p = 0.5$. The calibration-set accuracy is varied over $\theta_Q \in [0.25, 0.75]$, inducing label shift while keeping $\Pr(\hat{Z} | Z)$ unchanged. For each setting, we generate a test dataset of size 1000 and a calibration dataset of size 200 with $\Pr_Q(Z = 1) = \theta_Q$. Results are averaged over 10,000 Monte Carlo replications.

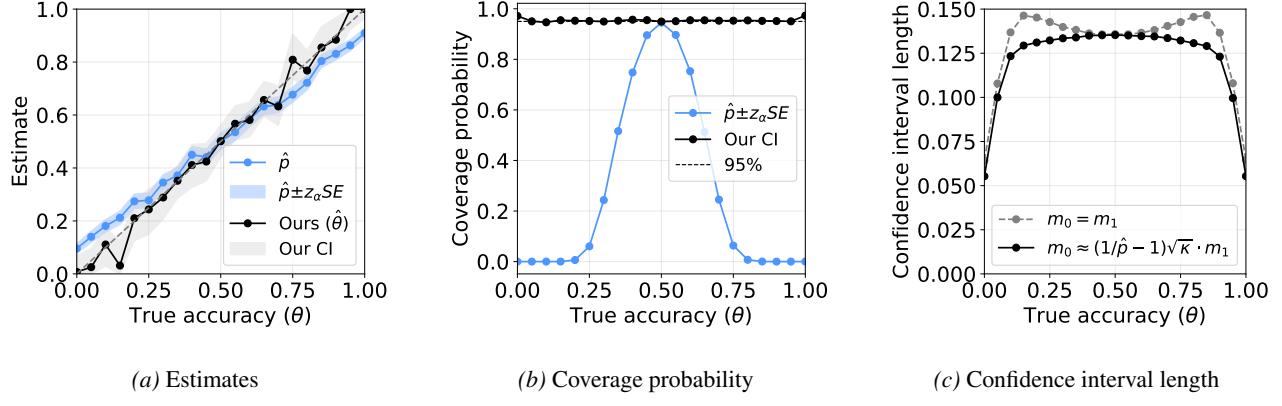
E. Additional Results on Monte Carlo Simulation

To complement the main simulation results presented in Figure 6, we report an extensive set of Monte Carlo experiments conducted across multiple configurations of the test dataset size $n = 1000$, the calibration sizes $m \in \{200, 500\}$, and the judge reliability parameters $(q_0, q_1) \in \{(0.9, 0.9), (0.7, 0.7), (0.9, 0.7), (0.7, 0.9)\}$. The remaining aspects of the simulation design follow the same setup as in the main text.

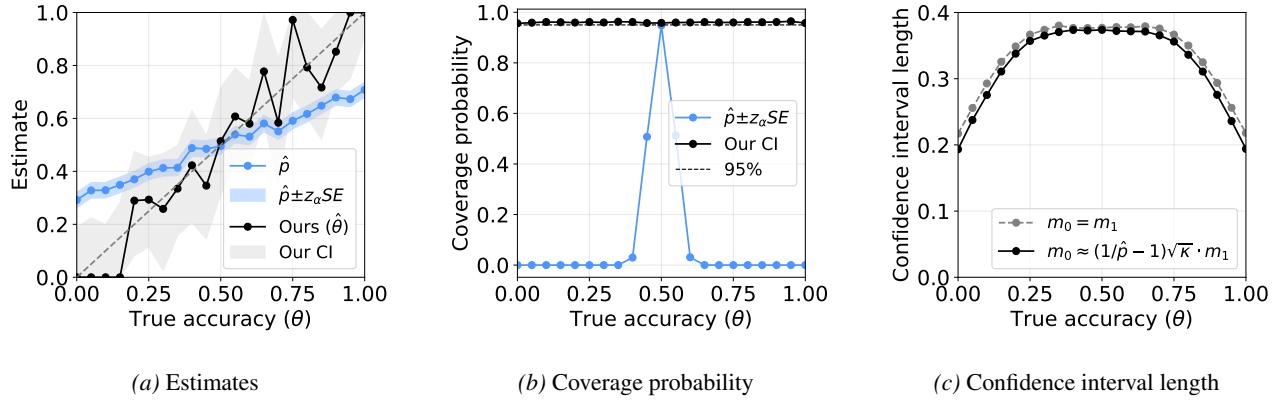
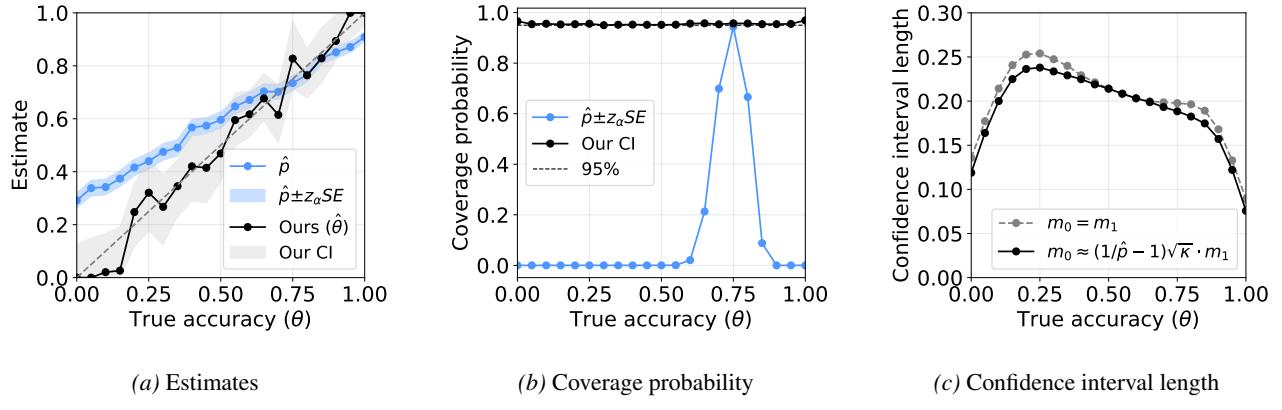
Across all combinations of (n, m, q_0, q_1) , the qualitative findings observed in the main simulation persist. Bias correction consistently improves estimation accuracy, empirical coverage attains the nominal level, and optimized calibration allocation yields shorter confidence intervals.

Below we present the complete collection of results. Each figure corresponds to one configuration (n, m, q_0, q_1) and includes three subplots.

E.1. Results for $n = 1000$ and $m = 200$

 Figure 9. Monte Carlo results for $(n, m, q_0, q_1) = (1000, 200, 0.7, 0.7)$.

 Figure 10. Monte Carlo results for $(n, m, q_0, q_1) = (1000, 200, 0.7, 0.9)$.

 Figure 11. Monte Carlo results for $(n, m, q_0, q_1) = (1000, 200, 0.9, 0.7)$.


 Figure 12. Monte Carlo results for $(n, m, q_0, q_1) = (1000, 200, 0.9, 0.9)$.

E.2. Results for $n = 1000$ and $m = 500$


 Figure 13. Monte Carlo results for $(n, m, q_0, q_1) = (1000, 500, 0.7, 0.7)$.

 Figure 14. Monte Carlo results for $(n, m, q_0, q_1) = (1000, 500, 0.7, 0.9)$.

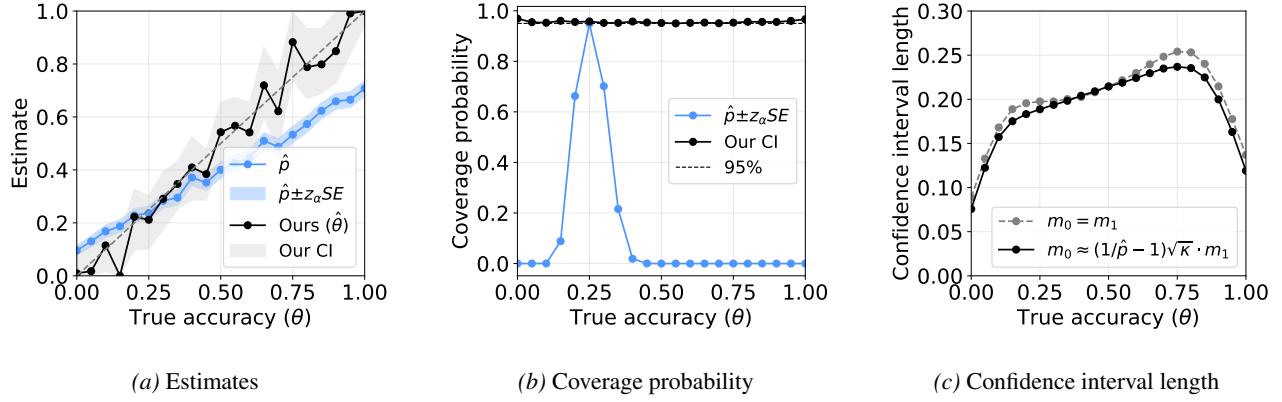


Figure 15. Monte Carlo results for $(n, m, q_0, q_1) = (1000, 500, 0.9, 0.7)$.

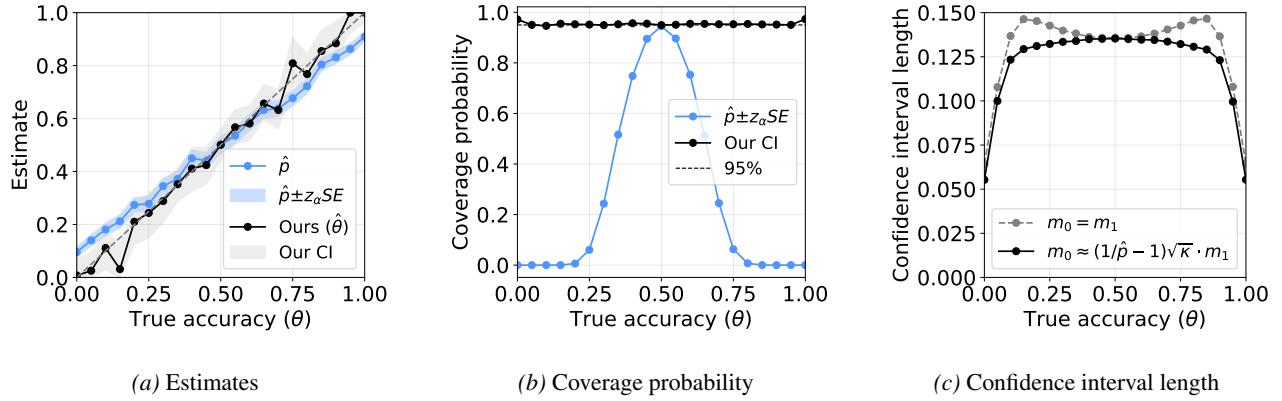


Figure 16. Monte Carlo results for $(n, m, q_0, q_1) = (1000, 500, 0.9, 0.9)$.