Correlation-Aware Ranking and Filtering for Heterogeneous Small-Model Judge Ensembles

Anonymous Author(s)
Affiliation
Address
email

Abstract

Small language model judges are often selected by accuracy alone, but majority voting can help only when selected judges are individually competent and do not fail on the same examples. We introduce Judge CIA (Correlation-Informed Agents), a selection framework that trains heterogeneous LoRA judge adapters and constructs juries by using held-out data to approximate competence and bounded-redundancy conditions. Our analysis gives an independent Condorcet-style bound, a bounded-redundancy diagnostic with variance-equivalent effective jury size, and a slice-wise decomposition for high-stakes risk. Across 12 RAGBench model–dataset settings, Judge CIA voting is non-worse than the strongest held-out-selected adapter in 10 settings by point estimate, with an average signed accuracy difference of $+1.85$. On Gemma-3-1B/TAT-QA, adding the correlation filter improves full-jury accuracy from 59.91% to 64.22% while reducing mean selected-pair phi correlation from 0.311 to 0.025. On a compliance dataset, slice-weighted ranking reduces average target-slice and critical-slice error proxies in the selected setting, while revealing heterogeneous effects across slices.

1 Introduction

Small language models (SLLMs) are increasingly used as task-specific judges for retrieval support, policy compliance, and domain-specific audits. A common approach is to fine-tune several adapters and deploy the one with the best accuracy. This can be brittle: adapters often share prompts, training data, architectures, and domain artifacts, so they may fail on the same examples. For judge deployment, especially in high-stakes settings, reliability therefore depends not only on individual accuracy but also on whether judge errors are sufficiently non-redundant.

We study Judge CIA (Correlation-Informed Agents), a selection framework for constructing small-model judge juries from heterogeneous LoRA adapters [10]. Judge CIA trains N adapters, filters adapters by competence and pairwise empirical error correlation, and aggregates the selected odd-sized jury by majority vote. The method is a feasibility filter rather than a scalar subset optimizer: it admits adapters only when selection statistics suggest the operating regime in which majority voting is expected to help.

Our analysis formalizes this regime. An independent Condorcet-style bound clarifies the competence condition for majority voting, while a bounded-redundancy diagnostic shows how correlated errors reduce the effective size of the jury. We also decompose high-stakes risk into slice prevalence, slice-wise error, and slice-specific amplification, motivating slice-weighted ranking when unweighted accuracy is not the right deployment proxy.

Empirically, we evaluate Judge CIA on the RAGBench release [6] of TAT-QA [24], FinQA [2], and PubMedQA [11], with an internally constructed compliance dataset for the high-stakes slice study.

Submitted to 40th Conference on Neural Information Processing Systems (NeurIPS 2026). Do not distribute.

Across 12 public model–dataset settings, Judge CIA voting is non-worse than the strongest selected adapter in 10 settings by point estimate and improves by $+1.85$ percentage points on average. In the Gemma-3-1B [8]/TAT-QA diagnostic, adding the correlation filter improves full-jury accuracy from 59.91% to 64.22% while reducing mean selected-pair error correlation from 0.311 to 0.025. These results support the central message: useful small-model judge ensembles should be selected for both competence and non-redundant errors, not accumulated by size alone.

2 Related work

Judge CIA builds on classical ensemble learning, where gains depend on both individual competence and non-redundant errors [4, 7]. Prior work studies diversity measures such as pairwise correlation, disagreement, and double-fault statistics [12], as well as pruning large ensembles into compact high-performing subsets [15]. Our contribution is not a new diversity statistic, but a specialization of these ideas to small-model judge ensembles with feasibility filtering, correlation diagnostics, and high-stakes slices.

LLM-based evaluators are widely used for scalable evaluation of generated text and assistant behavior [13, 23], and recent work replaces a single large judge with a heterogeneous panel of LLM evaluators to improve agreement with human judgments, reduce intra-model bias, and lower evaluation cost [19]. RAGBench provides explainable retrieval-augmented generation evaluation data [6]. We focus on small, fine-tuned judge adapters and on selecting juries with held-out competence and error-correlation filters rather than using a fixed panel composition. For high-stakes deployment, we also draw on cost-sensitive and slice-aware evaluation, where average accuracy may not match operational loss [5].

3 Judge CIA jury and correlation-aware filtering

3.1 Jury definition and independent baseline

We first present an idealized independence baseline. The theorem below is not assumed to hold for LoRA adapters, but clarifies why the selection rule first filters for individually competent judges. We begin with a tail bound for abstract binary random variables.

Theorem 3.1 (Hoeffding–Chernoff tail bound)**.** *Let $Z_1, \ldots, Z_n \in \{0, 1\}$ be independent random variables, and define*

$$S := \sum_{i=1}^{n} Z_i, \qquad \bar{\varepsilon} := \frac{1}{n} \sum_{i=1}^{n} \mathbb{E}[Z_i], \qquad \Delta := \tfrac{1}{2} - \bar{\varepsilon}.$$

If $\bar{\varepsilon} < 1/2$, then

$$\mathbb{P}(S \geq n/2) \leq \exp\bigl(-2n\Delta^2\bigr).$$

The proof is given in Appendix B.

Definition 3.2 (Judge CIA jury)**.** *Let $\mathcal{X}$ be an input space, $\mathcal{D}$ a data distribution on $\mathcal{X}$, and $f : \mathcal{X} \to \{0, 1\}$ the ground-truth oracle. An agent $A_i : \mathcal{X} \to \{0, 1\}$ is a randomized decision rule.*

Given an odd number n of agents $\{A_1, \ldots, A_n\}$ producing decisions $\hat{y}_i = A_i(x) \in \{0, 1\}$, the Judge CIA collective decision is the majority vote

$$\hat{y}(x) = \mathbf{1}\Bigl\{\sum_{i=1}^{n} (2\hat{y}_i - 1) > 0\Bigr\}.$$

The odd-cardinality convention avoids ties.

For the Judge CIA jury, define the agent error indicators and total error count by

$$Z_i := \mathbf{1}\{A_i(x) \neq f(x)\}, \qquad S := \sum_{i=1}^{n} Z_i,$$

Unless stated otherwise, probabilities are over $x \sim \mathcal{D}$ and any internal randomness of the agents, and we use Z_i for agent i's error indicator and $S = \sum_i Z_i$ for the total error count.

Corollary 3.3 (Average-error Condorcet-style bound for Judge CIA)**.** *If the agent error indicators $Z_1, \ldots, Z_n$ are independent and their average marginal error satisfies*

$$\bar{\varepsilon} := \frac{1}{n} \sum_{i=1}^{n} \mathbb{P}(A_i(x) \neq f(x)) < \tfrac{1}{2},$$

then the Judge CIA majority vote satisfies

$$\mathbb{P}(\hat{y}(x) \neq f(x)) \leq \exp(-2n\Delta^2) \quad \textit{for the majority margin } \Delta := \tfrac{1}{2} - \bar{\varepsilon}.$$

In particular, $\mathbb{P}(\hat{y}(x) \neq f(x)) \to 0$ as $n \to \infty$ whenever $\Delta > 0$ is fixed.

The proof is given in Appendix B. This is a heterogeneous, average-error version of the classical *Condorcet Jury Theorem*: independence and an aggregate margin $\Delta > 0$ make majority error decay, even when the individual marginal errors are not identical [22].

3.2 Error dependence and bounded redundancy

Independence is an idealized regime. In practice, adapters share architectures, training data, prompts, and domain artifacts, so their errors can be dependent. Judge CIA therefore does not try to enforce independence, common marginal error rates, or exact equicorrelation. Instead, it targets a weaker selection-relevant regime: each selected adapter should be individually competent, and selected pairs should not have overly redundant sample-wise error patterns. This aligns with classic ensemble theory that aggregation can outperform a single expert when errors are not perfectly correlated [4, 7].

Assumption 3.4 (Bounded competence and bounded redundancy)**.** *For a selected jury $\mathcal{K}$ with $|\mathcal{K}| = k$, suppose each selected judge satisfies*

$$\mathbb{P}(Z_i = 1) \leq \varepsilon_0 < 1/2 \quad (i \in \mathcal{K}),$$

and each selected pair satisfies

$$\mathrm{Corr}(Z_i, Z_j) \leq \phi \quad (i \neq j,\ i, j \in \mathcal{K}),$$

for some $0 \leq \phi \leq 1$. In practice, the selection rule in Section 5.3 approximates these population conditions using validation estimates: empirical error below $1 - \tau$ and empirical pairwise error correlation below a user-specified filtering threshold ρ.

Proposition 3.5 (Diagnostic majority bound under bounded redundancy)**.** *Let $S_{\mathcal{K}} := \sum_{i \in \mathcal{K}} Z_i$ and $\Delta_0 := 1/2 - \varepsilon_0$. Under Assumption 3.4,*

$$\mathrm{Var}(S_{\mathcal{K}}) \leq k\varepsilon_0(1 - \varepsilon_0)\big(1 + (k-1)\phi\big),$$

and Cantelli's inequality gives the diagnostic bound

$$\mathbb{P}(S_{\mathcal{K}} \geq k/2) \leq \frac{\varepsilon_0(1-\varepsilon_0)\big(1+(k-1)\phi\big)}{\varepsilon_0(1-\varepsilon_0)\big(1+(k-1)\phi\big) + k\Delta_0^2} = \frac{\varepsilon_0(1-\varepsilon_0)}{\varepsilon_0(1-\varepsilon_0) + k_{\mathrm{eff}}(\phi)\Delta_0^2}$$

with

$$k_{\mathrm{eff}}(\phi) := \frac{k}{1 + (k-1)\phi}.$$

The proof is given in Appendix B. This is a conservative diagnostic upper bound, not a tight prediction of majority-vote error. Appendix A gives the corresponding redundancy interpretation.

To explain one source of error correlation, let θ denote a common condition affecting all agents, such as a distribution shift, shared prompt artifact, or shared training-data bias. Agent i responds through its error propensity $p_i(\theta)$. If

$$Z_i \mid \theta \ \sim\ \mathrm{Bernoulli}(p_i(\theta)), \qquad Z_i \perp Z_j \mid \theta\ (i \neq j).$$

then conditional independence holds only after θ is fixed; marginally, agents may still be correlated because their propensities depend on the same θ. By the law of total covariance,

$$\mathrm{Cov}(Z_i, Z_j) = \mathrm{Cov}\big(p_i(\theta), p_j(\theta)\big) \quad (i \neq j),$$

so heterogeneity helps when it makes the propensity functions $p_i(\theta)$ and $p_j(\theta)$ less aligned. In our experiments, heterogeneity is induced by varying prompt template, LoRA rank/scaling settings, target-module set, and sampling seed. These axes are intended to produce adapters whose error propensities respond differently to shared latent difficulty factors, thereby reducing alignment among the $p_i(\theta)$ functions. We use Pearson correlation on binary error-indicator vectors, equivalently the phi correlation coefficient, as the empirical redundancy statistic [12]. Empirically, we estimate pairwise error correlations $\widehat{\rho}_{ij}$ from adapter error vectors and report mean and maximum selected-jury pairwise error correlation as scalar redundancy diagnostics. Pairwise correlations involving zero-variance error vectors are handled using the convention described in Appendix D.1.

4 Catastrophic risk

Majority aggregation can reduce error, but high-stakes deployment also depends on which errors are operationally severe. The selection criterion should therefore reflect deployment-relevant slice risk rather than average error alone [3]. Throughout, a slice denotes a measurable subset of examples defined by task metadata, operational context, or a high-stakes attribute. In the catastrophic-risk decomposition below, $\mathcal{H} \subseteq \mathcal{X}$ is the high-stakes subset and $(\mathcal{H}_s)_{s\in\mathcal{S}}$ denotes a disjoint measurable partition of $\mathcal{H}$. If monitored metadata slices overlap in an application, the decomposition can be applied to a disjoint refinement of those slices. In experiments, compliance slices are defined by criterion criticality, policy flags, and violation or failure-type metadata; details are given in Appendix C.

Definition 4.1 (Catastrophic event and risk)**.** *A* catastrophic event *is specified by a predicate*

$$\mathcal{C}(x, \hat{y}(x), f(x)) \in \{0, 1\},$$

where $\mathcal{C} = 1$ denotes that the realized decision induces a domain-specified severe adverse consequence. The predicate may depend on the input, the decision, and the ground truth. For $x \sim \mathcal{D}$, define the catastrophic risk of a decision rule $\hat{y}$ as

$$\mathrm{CatRisk}(\hat{y}) := \mathbb{P}\big(\mathcal{C}(x, \hat{y}(x), f(x)) = 1 \ \wedge \ \hat{y}(x) \neq f(x) \ \wedge \ x \in \mathcal{H}\big).$$

Here, $\mathcal{C}$ identifies the adverse consequence itself, while the additional conditions $\hat{y}(x) \neq f(x)$ and $x \in \mathcal{H}$ restrict evaluation to high-stakes errors. Examples include falsely approving a non-compliant consultation or falsely flagging a compliant consultation as severe.

Definition 4.2 (Slice-wise catastrophic amplification)**.** *Let $(\mathcal{H}_s)_{s\in\mathcal{S}}$ be the high-stakes slice partition above, with $\mathbb{P}(x \in \mathcal{H}_s) > 0$. Define the slice-wise misclassification rate*

$$\mathrm{Err}_s(\hat{y}) := \mathbb{P}\big(\hat{y}(x) \neq f(x) \mid x \in \mathcal{H}_s\big),$$

and, for a fixed decision rule $\hat{y}$, the slice-wise catastrophic amplification factor

$$\kappa_s(\hat{y}) := \mathbb{P}\big(\mathcal{C}(x, \hat{y}(x), f(x)) = 1 \mid x \in \mathcal{H}_s \ \wedge \ \hat{y}(x) \neq f(x)\big) \in [0, 1].$$

When the conditioning event has probability zero, we set $\kappa_s(\hat{y}) := 0$. Thus $\kappa_s(\hat{y})$ is the decision-rule-dependent probability that an error *becomes* catastrophic *on slice $\mathcal{H}_s$.*

Lemma 4.3 (Slice-specific catastrophic risk decomposition)**.**

$$\mathrm{CatRisk}(\hat{y}) = \sum_{s\in\mathcal{S}} \mathbb{P}(x \in \mathcal{H}_s)\, \kappa_s(\hat{y})\, \mathrm{Err}_s(\hat{y}).$$

Thus reducing catastrophic risk requires controlling slice-wise *error rates and, when possible, reducing the decision-rule-dependent amplification factor $\kappa_s(\hat{y})$ (changing the decision policy on that slice), not merely reducing average error.*

The proof is given in Appendix B.

Remark 4.4 (Why this is different from misclassification)**.** *Even when average error is small,* CatRisk *can be dominated by rare, high-impact slices. The decomposition separates slice-wise error control from the domain policy question of whether an error becomes catastrophic.*

5 Filtering and Slice-Weighted Ranking

Lemma 4.3 identifies slice-wise error as the part of the high-stakes risk proxy that can be addressed by agent selection. Building on the slice terminology in Section 4, we call $\mathcal{H}_s$ the target slice when it is assigned larger weight during slice-weighted ranking. The target-slice weighting protocol is detailed in Appendix D.4. We first formalize the statistics used to rank adapters: ordinary error and slice-weighted scores that emphasize high-stakes regions.

5.1 Ranking stability

Definition 5.1 (Selection sample and empirical error)**.** *Let $V := \{x^{(1)}, \dots, x^{(m)}\}$ be an i.i.d. selection sample set with $x^{(k)} \sim \mathcal{D}$. For agent A_i, define its true statistical error and empirical error by*

$$\varepsilon_i := \mathbb{P}(A_i(x) \neq f(x)), \qquad \hat{\varepsilon}_i := \frac{1}{m}\sum_{k=1}^{m} \mathbf{1}\{A_i(x^{(k)}) \neq f(x^{(k)})\}.$$

Equivalently, $\varepsilon_i = \mathbb{E}[Z_i]$ without assuming a common error rate across agents.

Definition 5.2 (Top-k ranking rule)**.** *Fix an integer $k \in \{1, \dots, n\}$. Compute empirical errors $\hat{\varepsilon}_1, \dots, \hat{\varepsilon}_n$ on the selection sample set and keep the k agents with the smallest empirical error. Formally, let $\hat{\sigma}$ be a permutation such that*

$$\hat{\varepsilon}_{\hat{\sigma}(1)} \leq \hat{\varepsilon}_{\hat{\sigma}(2)} \leq \cdots \leq \hat{\varepsilon}_{\hat{\sigma}(n)}.$$

Define the kept set

$$\widehat{\mathcal{K}}_k := \{\hat{\sigma}(1), \dots, \hat{\sigma}(k)\}.$$

Proposition 5.3 (Ranking stability of top-k under a separation margin)**.** *Let σ be a permutation that sorts the* true *errors:*

$$\varepsilon_{\sigma(1)} \leq \varepsilon_{\sigma(2)} \leq \cdots \leq \varepsilon_{\sigma(n)}, \qquad \mathcal{K}_k := \{\sigma(1), \dots, \sigma(k)\}.$$

Assume a separation margin at the boundary:

$$\varepsilon_{\sigma(k)} \leq \varepsilon_{\sigma(k+1)} - 2\gamma \quad \textit{for some } \gamma > 0.$$

Then the empirical top-k ranking recovers the true top-k set with probability at least

$$\mathbb{P}\big(\widehat{\mathcal{K}}_k = \mathcal{K}_k\big) \geq 1 - 2n\exp(-2m\gamma^2).$$

Equivalently, with probability at least $1 - 2n\exp(-2m\gamma^2)$,

$$\max_{i \in \mathcal{K}_k} \hat{\varepsilon}_i \leq \min_{j \notin \mathcal{K}_k} \hat{\varepsilon}_j.$$

The proof is given in Appendix B.

Risk note. This stability result controls only ranking error on the selection sample distribution. If V under-samples operationally important sub-slices (especially high-stakes or rare slices), then the recovered ranking may be optimal for $\mathcal{D}$ yet suboptimal for deployment. In high-stakes settings, V should be stratified to include the catastrophic slices defined in Section 4.

5.2 Slice-weighted ranking

Definition 5.4 (Slice-weighted empirical error)**.** *Let $(\mathcal{H}_s)_{s \in \mathcal{S}}$ be a measurable partition of a high-stakes subset $\mathcal{H} \subseteq \mathcal{X}$ (as in Section 4). Let $w : \mathcal{X} \to [0, W]$ be a bounded weight function that upweights high-stakes slices; for example, one may use*

$$w(x) = 1 + \sum_{s \in \mathcal{S}} (w_s - 1)\mathbf{1}\{x \in \mathcal{H}_s\}, \qquad 1 \leq w_s \leq W.$$

Given an i.i.d. selection sample set $V = \{x^{(1)}, \dots, x^{(m)}\}$ with $x^{(k)} \sim \mathcal{D}$, define the slice-weighted empirical error of agent A_i as

$$\hat{\varepsilon}_i^{(w)} := \frac{\sum_{k=1}^{m} w(x^{(k)})\, \mathbf{1}\{A_i(x^{(k)}) \neq f(x^{(k)})\}}{\sum_{k=1}^{m} w(x^{(k)})}.$$

For ranking agents, we use the equivalent unnormalized weighted score

$$\widehat{L}_i^{(w)} := \frac{1}{m}\sum_{k=1}^{m} w(x^{(k)})\,\mathbf{1}\{A_i(x^{(k)}) \neq f(x^{(k)})\}.$$

This score induces the same ordering as $\hat{\varepsilon}_i^{(w)}$ whenever the common denominator is positive, and remains well-defined when the empirical denominator is zero. Define the corresponding weighted true *error*

$$\varepsilon_i^{(w)} := \frac{\mathbb{E}_{x\sim\mathcal{D}}[w(x)\,\mathbf{1}\{A_i(x) \neq f(x)\}]}{\mathbb{E}_{x\sim\mathcal{D}}[w(x)]}.$$

Definition 5.5 (Top-k ranking with slice-weighted empirical error)**.** *Replace $\hat{\varepsilon}_i$ in the top-k ranking rule by $\widehat{L}_i^{(w)}$ and keep the k agents with the smallest weighted empirical scores. Denote the kept set by $\widehat{\mathcal{K}}_k^{(w)}$.*

Proposition 5.6 (Ranking stability of weighted top-k under a separation margin)**.** *Let σ be a permutation that sorts the* weighted true *errors:*

$$\varepsilon_{\sigma(1)}^{(w)} \le \varepsilon_{\sigma(2)}^{(w)} \le \cdots \le \varepsilon_{\sigma(n)}^{(w)}, \qquad \mathcal{K}_k^{(w)} := \{\sigma(1), \ldots, \sigma(k)\}.$$

Assume a separation margin at the boundary:

$$\varepsilon_{\sigma(k)}^{(w)} \le \varepsilon_{\sigma(k+1)}^{(w)} - 2\gamma \quad \textit{for some } \gamma > 0.$$

Assume further that the weight function is bounded as $w(x) \in [0, W]$ and that the mean weight *is uniformly bounded below:*

$$\mu_w := \mathbb{E}_{x\sim\mathcal{D}}[w(x)] \ge \mu_{\min} > 0.$$

Then the weighted empirical top-k ranking recovers the true weighted top-k set with probability at least

$$\mathbb{P}\big(\widehat{\mathcal{K}}_k^{(w)} = \mathcal{K}_k^{(w)}\big) \ge 1 - 2n\exp\Big(-\frac{2m\,\mu_{\min}^2\,\gamma^2}{W^2}\Big).$$

The proof is given in Appendix B.

Remark 5.7 (Direct control of slice-wise error)**.** *Choosing $w(x)$ concentrated on high-stakes slices makes top-k selection prioritize lower slice-weighted errors. In particular, if $w(x) = \sum_s w_s \mathbf{1}\{x \in \mathcal{H}_s\}$, then ranking by $\varepsilon_i^{(w)}$ emphasizes the slices with larger w_s. This is the relevant proxy statistic when high-stakes risk is dominated by specific slices, since* $\mathrm{CatRisk}$ *decomposes through slice-wise error terms rather than through average error alone.*

Domain routing can be incorporated by conditioning the catastrophic-risk decomposition on the true and predicted domains. We give the corresponding router–jury–slice decomposition in Appendix F.

5.3 Correlation-aware selection rule

We now combine the ranking scores above with the redundancy statistic from Section 3. Given a ranking score r_i for each adapter, where r_i is either the ordinary error $\hat{\varepsilon}_i$ or the slice-weighted score $\widehat{L}_i^{(w)}$, Judge CIA first forms the competent candidate pool

$$P_\tau = \{i : \widehat{\mathrm{Acc}}_i \ge \tau\}.$$

Here $\widehat{\mathrm{Acc}}_i = 1 - \hat{\epsilon}_i$, so the threshold τ is equivalently an empirical-error threshold $1 - \tau$. It then traverses P_τ in increasing order of r_i and accepts an adapter only if its empirical error correlation with every previously accepted adapter is at most ρ:

$$i \text{ is accepted only if } \max_{j\in\widehat{K}} \widehat{\rho}_{ij} \le \rho.$$

The resulting set $\widehat{K}$ is therefore an approximation to the competence and bounded-redundancy conditions in Assumption 3.4. The rule is intentionally a feasibility filter rather than a fixed-size subset optimizer: it preserves high-ranked competent adapters whose error patterns are not too redundant with the adapters already selected.

Fixed-size juries use the first k accepted adapters from this traversal, while the main voting results use the full feasible set, with the lowest-ranked selected adapter removed only to ensure odd cardinality. Slice weighting changes only the ranking score, not the unweighted competence threshold τ. Appendix D.1 gives pseudocode for this rule as Algorithm 1.

Table 1: Main performance across SLLMs and RAGBench datasets. Δ denotes the signed accuracy difference $\mathrm{Vote} - \mathrm{Top\text{-}1}$; positive values favor Vote. Values use the same percentage scale as the accuracy columns. Model-level average rows summarize Δ across the three datasets; confidence intervals are reported in Appendix D.2.

	FinQA				TAT-QA				PubMedQA				
Model	Vanilla	Top-1	Vote	Δ	Vanilla	Top-1	Vote	Δ	Vanilla	Top-1	Vote	Δ	Avg. Δ
Qwen3-0.6B	49.49	56.38	61.22	4.85	43.97	61.21	58.62	-2.59	16.00	61.20	61.20	0.00	0.75
Qwen3-1.7B	48.72	65.31	67.09	1.79	45.69	66.81	68.10	1.29	35.60	59.60	62.20	2.60	1.89
Llama-3.2-1B	54.08	57.40	57.40	0.00	53.02	63.36	62.50	-0.86	50.80	57.20	60.40	3.20	0.78
Gemma-3-1B	50.00	52.81	53.32	0.51	45.26	59.05	64.22	5.17	50.40	52.60	58.80	6.20	3.96
Average	50.57	57.97	59.76	1.79	46.99	62.61	63.36	0.75	38.20	57.65	60.65	3.00	1.85

6 Experiments

The SLLMs are Qwen3-0.6B and Qwen3-1.7B [21], Llama-3.2-1B [16], and Gemma-3-1B [8]. We evaluate majority-vote improvements, pairwise error correlation, jury-size curves, ranking stability, and high-stakes slice weighting. We use RAGBench TAT-QA, FinQA, and PubMedQA for the main experiments [6], and an internally constructed telephone-consultation compliance dataset for the high-stakes study.

We organize the experiments around four hypotheses: **H1** heterogeneous adapter construction lowers pairwise phi correlation; **H2** among competent adapters, phi-correlation filtering improves majority-vote behavior relative to accuracy-only selection; **H3** filtering and ranking decisions become more stable as selection sample size increases; and **H4** slice-weighted ranking can reduce target-slice error, possibly trading off non-target critical-slice performance.

6.1 Experimental setup

For each base SLLM, we train a heterogeneous pool of LoRA adapters [10] on the training split, select adapters on the validation split, and evaluate single-adapter and majority-vote decisions on the test split. In this experimental protocol, the selection sample set V from Section 5 is the validation split. Detailed hyperparameters, adapter-pool dimensions, and additional comparison tables are in Appendix D–E. Because our main question is how to select among heterogeneous small-adapter judges from the same candidate pool, we keep single-backbone comparison results in Appendix E as external context rather than treating them as mechanism-level baselines.

6.2 Main performance

We first ask whether a selected heterogeneous jury improves over the validation-selected Top-1 adapter in the candidate pool. Table 1 compares the unfine-tuned base model (Vanilla), the strongest validation-selected adapter (Top-1), and the validation-selected correlation-aware jury (Vote) across four SLLMs and three RAGBench datasets. Section 6.4 isolates the contribution of the correlation filter by comparing this rule against random and accuracy-only jury construction.

Across the 12 model–dataset cells, voting is usually competitive with or better than the validation-selected Top-1 adapter, but the gains are not uniform. Gemma-3-1B shows the clearest benefit, while two TAT-QA settings lose slightly relative to Top-1. This pattern is consistent with the theory: majority voting is useful only when the selected agents are both competent and non-redundant, so Table 1 should be read together with the diagnostics below.

6.3 Heterogeneity and error correlation

H1 tests the prediction from Section 3 that majority voting is most useful when agent errors are not strongly correlated. We test whether heterogeneous adapter design reduces shared failure modes and shifts the pairwise error-correlation distribution downward. Figure 1 compares a heterogeneous Qwen3-1.7B/TAT-QA pool against a fixed-axis pool that retains adapter-level training variation but holds the main design choices fixed. The heterogeneous pool shifts the pairwise error-correlation

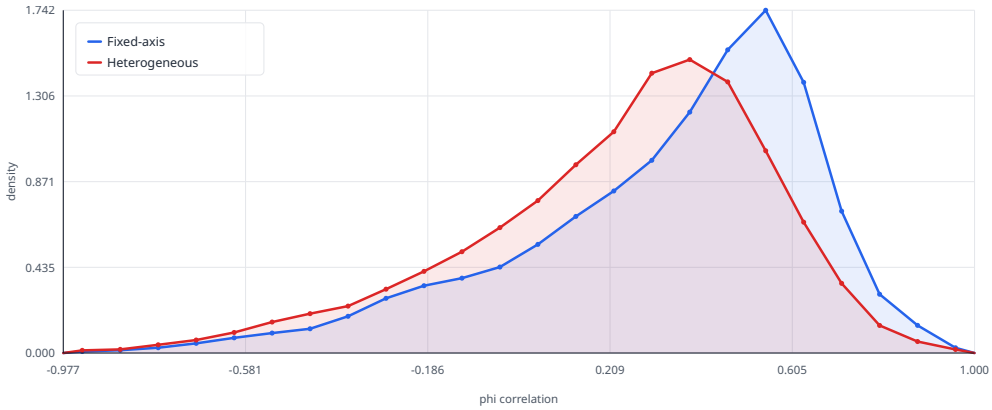


Figure 1: Pairwise error-correlation distributions. Fixed-axis varies only the seed, while heterogeneous varies prompt, LoRA rank/scale settings, target-module, and seed choices.

Table 2: Pool-selection comparison across random, accuracy-only, and correlation-aware jury construction. The Random row averages over 50 random draws from the competent-adapter pool. Accuracy differences are measured relative to Top-1 on the same percentage scale as full-jury accuracy. Lower pairwise error correlation is associated with the strongest majority-vote result. Here, $\hat{\rho}$ denotes empirical phi correlation between selected adapters' error-indicator vectors.

Selection rule	Full-pool acc. (%)	Jury size	Accuracy difference	Mean $\hat{\rho}$	Max $\hat{\rho}$
Random	57.84	23	-1.22	0.087	0.846
Accuracy-only	59.91	23	0.86	0.311	0.761
Accuracy+corr.	64.22	23	5.17	0.025	0.398

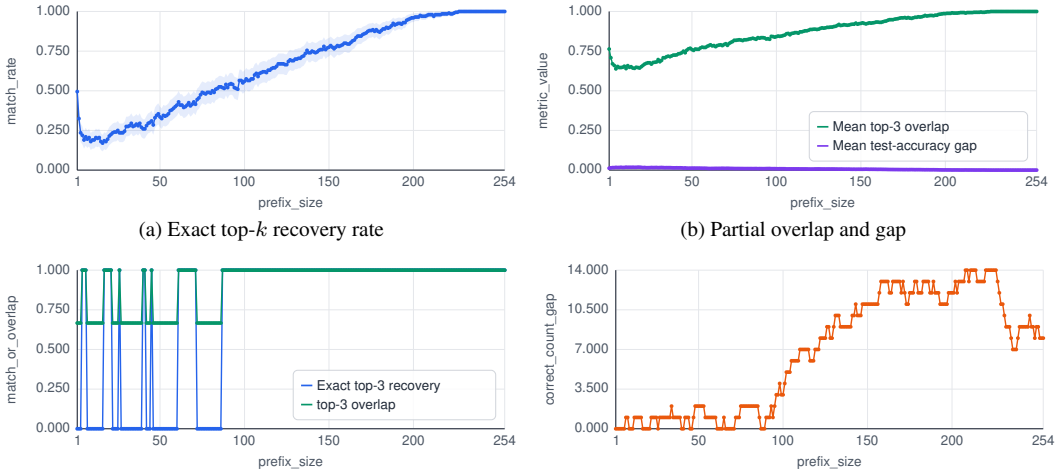
distribution downward, with lower mean and median phi correlations and more low-correlation pairs. Thus H1 is supported: varying prompt and LoRA design axes produces visibly less redundant adapter errors than varying only training randomness.

6.4 Correlation versus jury error

H2 asks whether lower empirical correlation translates into lower majority-vote error. This subsection explains the mechanism behind Table 1 rather than replacing the headline performance comparison. On TAT-QA with Gemma-3-1B, we compare three ways to form juries: random selection from competent adapters averaged over 50 draws (Random), validation-accuracy ranking without the correlation filter (Accuracy-only), and the proposed rule with the additional pairwise error-correlation filter (Accuracy+corr.). Table 2 summarizes the full-pool comparison, and Appendix D.3 shows the full jury-size curve. Thus Table 1 reports the main accuracy outcome, while this diagnostic comparison tests whether the strongest gains are associated with fewer shared error patterns. The rule with the additional correlation filter has fewer shared error patterns in the selected jury and the best full-pool test accuracy among the three rules. These results support H2 for this diagnostic setting and explain why the Gemma-3-1B/TAT-QA cell in Table 1 is one of the strongest majority-vote gains.

6.5 Filtering and ranking stability

H3 asks whether filtering and ranking decisions become more stable as validation size increases. Motivated by Proposition 5.3, Figure 2 recomputes the top-k adapter set on growing validation prefixes and compares it with a post hoc full-test top-k proxy. This diagnostic is run on the Qwen3-0.6B/TAT-QA adapter pool, with the test-set proxy used only for analysis and never for adapter selection. Small prefixes are unstable, but the prefix-selected set increasingly overlaps with the full-test proxy as more validation data are used. This supports H3 qualitatively and shows why the procedure should be treated as validation-size dependent rather than as a fixed property of the adapter pool.



(a) Exact top-k recovery rate

(b) Partial overlap and gap

(c) Original-order alignment

(d) Boundary gap

Figure 2: Prefix-alignment diagnostics for validation-based filtering and ranking. The reference set is the post hoc full-test top-k proxy. Panels (a) and (b) aggregate over shuffled validation-prefix orders: (a) reports the strict exact-recovery rate, while (b) reports partial top-k overlap and the test-accuracy gap to that proxy. Panel (c) shows the same recovery and overlap diagnostics in the original validation order; panel (d) reports the empirical boundary gap.

Table 3: High-stakes slice-weighted selection results. Columns report test error rates: All is the full test set, Target is the upweighted high-stakes slice, Other crit. pools the remaining critical slices, Critical is the full critical subset, and Target mass is the target-slice contribution to total test error. Rows aggregate over the five target-slice runs reported individually in Table 7.

	Setting		Test error rate (%)				
Rule	k	w_{target}	All	Target	Other crit.	Critical	Target mass
Unweighted top-k	9	1	2.60	2.59	2.26	2.26	0.52
Slice-weighted top-k	9	1.5	2.48	2.38	1.70	1.65	0.48

6.6 High-stakes slice weighting

Lemma 4.3 shows that average error can be misleading when rare high-stakes slices dominate risk. **H4** evaluates whether slice-weighted validation ranking reduces target-slice error, possibly trading off non-target critical-slice performance. Since this dataset does not provide a fully observed catastrophic predicate, this experiment evaluates the controllable component of the decomposition: slice-wise error on operationally critical compliance slices. Using the internally constructed compliance dataset with Qwen3-1.7B, we compare average validation-error ranking with slice-weighted ranking on the same candidate pool. Table 3 reports overall error, target-slice error, error on other critical slices, full critical-subset error, and target-slice mass error. The slice-weighted rule reduces the target-slice proxy without increasing average error in the selected setting. Appendix D.4 shows that this effect is heterogeneous across slices, so H4 is partially supported: slice weighting can move the jury toward high-stakes slices, but it should be monitored slice by slice rather than summarized only by the average.

7 Conclusion

Judge CIA builds small-model judge juries by selecting for the two properties that majority voting requires but accuracy alone does not capture: competence and non-redundant errors. The theory explains this dependence through an independent majority bound and a bounded-redundancy diagnostic, while the experiments show that correlation-aware voting is usually competitive with or better than the strongest held-out-selected adapter. The high-stakes study further shows that slice-weighted ranking can target operationally important error proxies when average accuracy is insufficient.

References

[1] Marah Abdin, Jyoti Aneja, Harkirat Behl, Sebastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J. Hewett, Mojan Javaheripi, Piero Kauffmann, James R. Lee, Yin Tat Lee, et al. Phi-4 technical report, 2024.

[2] Zhiyu Chen, Wenhu Chen, Charese Smiley, Sameena Shah, Iana Borova, Dylan Langdon, Reema Moussa, Matt Beane, Ting-Hao Huang, Bryan R. Routledge, and William Yang Wang. Finqa: A dataset of numerical reasoning over financial data. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3697–3711, 2021.

[3] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley, July 2006.

[4] Thomas G. Dietterich. Ensemble Methods in Machine Learning. In Gerhard Goos, Juris Hartmanis, and Jan Van Leeuwen, editors, *Multiple Classifier Systems*, volume 1857, pages 1–15. Springer Berlin Heidelberg, Berlin, Heidelberg, 2000.

[5] Charles Elkan. The foundations of cost-sensitive learning. In *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence*, pages 973–978, 2001.

[6] Robert Friel, Masha Belyi, and Atindriyo Sanyal. Ragbench: Explainable benchmark for retrieval-augmented generation systems, 2025.

[7] Stuart Geman, Elie Bienenstock, and René Doursat. Neural networks and the bias/variance dilemma. *Neural computation*, 4(1):1–58, 1992.

[8] Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, et al. Gemma 3 technical report, 2025.

[9] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, et al. The llama 3 herd of models, 2024.

[10] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

[11] Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William W. Cohen, and Xinghua Lu. Pubmedqa: A dataset for biomedical research question answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 2567–2577, 2019.

[12] Ludmila I. Kuncheva and Christopher J. Whitaker. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Machine Learning*, 51(2):181–207, 2003.

[13] 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*, pages 2511–2522, 2023.

[14] Sourab Mangrulkar, Sylvain Gugger, Lysandre Debut, Younes Belkada, Sayak Paul, Benjamin Bossan, and Marian Tietz. PEFT: State-of-the-art parameter-efficient fine-tuning methods. `https://github.com/huggingface/peft`, 2022.

[15] Dragos D. Margineantu and Thomas G. Dietterich. Pruning adaptive boosting. In *Proceedings of the Fourteenth International Conference on Machine Learning*, pages 211–218, 1997.

[16] Meta Llama. Llama 3.2 model card. `https://github.com/meta-llama/llama-models/blob/main/models/llama3_2/MODEL_CARD.md`, 2024. Accessed: 2026-05-07.

[17] Mistral AI. Introducing mistral 3. `https://mistral.ai/news/mistral-3`, 2025. Accessed: 2026-05-07.

[18] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, 2019.

[19] 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, 2024.

[20] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, 2020.

[21] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, et al. Qwen3 technical report, 2025.

[22] H. Peyton Young. Condorcet's theory of voting. *American Political science review*, 82(4):1231–1244, 1988.

[23] 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.

[24] Fengbin Zhu, Wenqiang Lei, Youcheng Huang, Chao Wang, Shuo Zhang, Jiancheng Lv, Fuli Feng, and Tat-Seng Chua. Tat-qa: A question answering benchmark on a hybrid of tabular and textual content in finance. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, 2021.

A Equicorrelated Diagnostic Special Case

The main text uses Assumption 3.4 because it matches the filtering rule: selected adapters need only have bounded empirical error and bounded pairwise empirical error correlation. For intuition, the bounded-redundancy diagnostic reduces to the following common-error, equicorrelated special case. This appendix result is not an assumption enforced by Judge CIA.

Assumption A.1 (Equicorrelated diagnostic model)**.** *Assume a common marginal error* $\mathbb{E}[Z_i] = \varepsilon$ *and*

$$\mathrm{Corr}(Z_i, Z_j) = \rho \quad \textit{for all } i \neq j,$$

where ρ *lies in the admissible range for the joint Bernoulli distribution and* $1 + (n-1)\rho > 0$*. The main regime of interest is* $\rho \geq 0$*, where redundant errors reduce the usable jury size.*

Proposition A.2 (Equicorrelated effective jury size diagnostic)**.** *Under Assumption A.1, assume* $\varepsilon < 1/2$ *and let* $\Delta_{\mathrm{diag}} := 1/2 - \varepsilon$*. Define*

$$n_{\mathrm{eff}} := \frac{n}{1 + (n-1)\rho}.$$

Then $\mathrm{Var}(S) = n\varepsilon(1-\varepsilon)(1 + (n-1)\rho)$*, and*

$$\mathbb{P}(S \geq n/2) \leq \frac{\mathrm{Var}(S)}{\mathrm{Var}(S) + n^2\Delta_{\mathrm{diag}}^2} = \frac{\varepsilon(1-\varepsilon)}{\varepsilon(1-\varepsilon) + n_{\mathrm{eff}}\Delta_{\mathrm{diag}}^2}.$$

The same variance-equivalent reading applies to Proposition 3.5 in the main text. There, $k_{\mathrm{eff}}(\phi) = k/(1 + (k-1)\phi)$ is a diagnostic for how much pairwise redundancy inflates the variance of the jury error count. It should not be read as a calibrated prediction of test error or as an assumption that selected adapters have common error rates or exact equicorrelation.

B Proofs

Proof of Theorem 3.1. We have $\mathbb{E}[S] = n\bar{\varepsilon}$, so $\{S \geq n/2\} = \{S - \mathbb{E}[S] \geq n\Delta\}$. Since the $Z_i \in [0,1]$ are independent, Hoeffding's inequality gives

$$\mathbb{P}(S - \mathbb{E}[S] \geq n\Delta) \leq \exp(-2n\Delta^2).$$

□

Proof of Corollary 3.3. A majority-vote error can occur only when at least half of the agents err:

$$\{\hat{y}(x) \neq f(x)\} \subseteq \{S \geq n/2\}.$$

The result follows from Theorem 3.1. □

Proof of Proposition A.2. First,

$$\mathrm{Var}(S) = \sum_i \mathrm{Var}(Z_i) + 2\sum_{i<j} \mathrm{Cov}(Z_i, Z_j) = n\varepsilon(1-\varepsilon)(1+(n-1)\rho).$$

Cantelli's inequality gives

$$\mathbb{P}(X - \mu \geq t) \leq \frac{\sigma^2}{\sigma^2 + t^2}.$$

Apply it with $X = S$, $\mu = \mathbb{E}[S] = n\varepsilon$, $t = n\Delta_{\mathrm{diag}}$, and $\sigma^2 = \mathrm{Var}(S)$. □

Proof of Proposition 3.5. For each $i \in \mathcal{K}$, $\mathbb{P}(Z_i = 1) \leq \varepsilon_0 < 1/2$, so

$$\mathrm{Var}(Z_i) \leq \varepsilon_0(1-\varepsilon_0).$$

For $i \neq j$ in $\mathcal{K}$,

$$\mathrm{Cov}(Z_i, Z_j) \leq \phi\sqrt{\mathrm{Var}(Z_i)\mathrm{Var}(Z_j)} \leq \phi\,\varepsilon_0(1-\varepsilon_0).$$

Therefore

$$\mathrm{Var}(S_{\mathcal{K}}) \leq k\varepsilon_0(1-\varepsilon_0) + k(k-1)\phi\varepsilon_0(1-\varepsilon_0).$$

Also $\mathbb{E}[S_{\mathcal{K}}] \leq k\varepsilon_0$, hence

$$\{S_{\mathcal{K}} \geq k/2\} \subseteq \{S_{\mathcal{K}} - \mathbb{E}[S_{\mathcal{K}}] \geq k\Delta_0\}.$$

Applying Cantelli's inequality with the variance bound above gives the result. □

Proof of Lemma 4.3. For readability, write $\mathcal{C} = 1$ for $\mathcal{C}(x, \hat{y}(x), f(x)) = 1$. Since $(\mathcal{H}_s)_{s\in\mathcal{S}}$ partitions $\mathcal{H}$,

$$\mathrm{CatRisk}(\hat{y}) = \sum_{s\in\mathcal{S}} \mathbb{P}(\mathcal{C} = 1 \wedge \hat{y} \neq f \wedge x \in \mathcal{H}_s).$$

For a fixed slice s,

$$\mathbb{P}(\mathcal{C} = 1 \wedge \hat{y} \neq f \wedge x \in \mathcal{H}_s) = \mathbb{P}(x \in \mathcal{H}_s)\mathbb{P}(\mathcal{C} = 1 \wedge \hat{y} \neq f \mid x \in \mathcal{H}_s).$$

If $\mathrm{Err}_s(\hat{y}) = 0$, this term is zero. Otherwise, the chain rule gives

$$\begin{aligned}\mathbb{P}(\mathcal{C} = 1 \wedge \hat{y} \neq f \mid x \in \mathcal{H}_s) &= \mathbb{P}(\mathcal{C} = 1 \mid x \in \mathcal{H}_s \wedge \hat{y} \neq f)\,\mathbb{P}(\hat{y} \neq f \mid x \in \mathcal{H}_s) \\ &= \kappa_s(\hat{y})\,\mathrm{Err}_s(\hat{y}).\end{aligned}$$

Summing over $s \in \mathcal{S}$ gives the equality. □

Proof of Proposition 5.3. If $|\hat{\varepsilon}_i - \varepsilon_i| < \gamma$ holds for all $i \in \{1, \ldots, n\}$, then

$$\max_{i\in\mathcal{K}_k} \hat{\varepsilon}_i \leq \varepsilon_{\sigma(k)} + \gamma \leq \varepsilon_{\sigma(k+1)} - \gamma \leq \min_{j\notin\mathcal{K}_k} \hat{\varepsilon}_j.$$

Thus the empirical ordering cannot swap any element across the $k/(k+1)$ boundary, implying $\widehat{\mathcal{K}}_k = \mathcal{K}_k$. By the single-agent Hoeffding lemma and a union bound,

$$\mathbb{P}(\exists i : |\hat{\varepsilon}_i - \varepsilon_i| \geq \gamma) \leq 2n\exp(-2m\gamma^2),$$

which proves the claim. □

Proof of Proposition 5.6. Recall the numerator score

$$L_i^{(w)} := \mathbb{E}_{x\sim\mathcal{D}}[w(x)\,\mathbf{1}\{A_i(x) \neq f(x)\}].$$

The denominator in $\hat{\varepsilon}_i^{(w)}$ is common to all agents, so the weighted empirical-error ordering is identical to the ordering by $\widehat{L}_i^{(w)}$ whenever the denominator is positive. Similarly,

$$L_i^{(w)} = \mu_w\, \varepsilon_i^{(w)},$$

where $\mu_w = \mathbb{E}_{x\sim\mathcal{D}}[w(x)]$ is common to all agents, so the true ordering by $\varepsilon_i^{(w)}$ is identical to the true ordering by $L_i^{(w)}$. Each summand in $\widehat{L}_i^{(w)}$ lies in $[0, W]$, and Hoeffding's inequality gives

$$\mathbb{P}\big(|\widehat{L}_i^{(w)} - L_i^{(w)}| \geq \mu_w\gamma\big) \leq 2\exp\Big(-\frac{2m\,\mu_w^2\gamma^2}{W^2}\Big) \leq 2\exp\Big(-\frac{2m\,\mu_{\min}^2\gamma^2}{W^2}\Big).$$

Union bound over $i \in \{1, \ldots, n\}$ gives that, with probability at least

$$1 - 2n\exp\Big(-\frac{2m\,\mu_{\min}^2\gamma^2}{W^2}\Big),$$

we have $|\widehat{L}_i^{(w)} - L_i^{(w)}| < \mu_w\gamma$ for all i. The separation condition implies

$$L_{\sigma(k)}^{(w)} = \mu_w\varepsilon_{\sigma(k)}^{(w)} \leq \mu_w\varepsilon_{\sigma(k+1)}^{(w)} - 2\mu_w\gamma = L_{\sigma(k+1)}^{(w)} - 2\mu_w\gamma.$$

Thus the same boundary-swap argument as in the unweighted case implies that no agent can cross the $k/(k+1)$ boundary, hence $\widehat{\mathcal{K}}_k^{(w)} = \mathcal{K}_k^{(w)}$. □

C Compliance Dataset Details

The high-stakes dataset is an internally constructed telephone-consultation audit dataset for securities-domain compliance. It is generated by a script-based data-construction pipeline from normalized criterion items, violation templates, policy flags, and deterministic audit metadata. The canonical sample unit is a single-unit script audit centered on one normalized criterion item, rather than a full dialogue as the primary training unit. Each example contains dialogue turns, evidence turns, item labels, and metadata describing the intended compliance criterion and any injected fault. The full constructed split contains 4720 training rows, 590 validation rows, and 590 test rows, with balanced binary labels in each split. Following the common evaluation protocol in Appendix D, the experiments cap validation and test evaluation at 500 examples each.

The data specification defines high-stakes examples through criterion-level criticality and policy flags. Examples of policy flags include explicit-confirmation requirements, numerical-precision requirements, prohibitions on leading the customer's answer, waiting-for-answer requirements, and mandatory explanation-on-request rules. Violations are organized by operational fault types corresponding to coercive inducement, deceptive bypass, topic evasion, vulnerable-group handling, and fact omission. These fields induce slices that correspond to concrete compliance failures, which is why the dataset is used for the slice-wise high-stakes error proxy experiment in Section 6.6. For the slice-weighted experiment, validation and test results are stored as a single bundle containing sample metadata and adapter-level correctness vectors. Each metadata row records the high-stakes flag and slice label, and each adapter row stores a binary correctness vector aligned to the metadata order. Thus a value of 1 means that the adapter is correct on the corresponding sample, while 0 means that it is incorrect.

D Experimental Details

Table 4 reports hyperparameters shared by the RAGBench and internally constructed compliance experiments. The model and dataset choices are described in Section 6; this appendix records implementation-level settings needed to reproduce the adapter pools and selection pipeline.

Table 4: Training, evaluation, and selection hyperparameters used for the main experiments.

Stage	Setting	Value
Data	Training split	At most 1000 training examples per task
	Evaluation splits	At most 500 validation examples and 500 test examples per task
	Reported metric	Accuracy on balanced validation/test binary-label evaluation sets for RAGBench
Training	Adapter method	LoRA; dropout 0.05; bias disabled
	Optimization	1 epoch; learning rate 2×10^{-4}; per-device train batch size 4; gradient accumulation 4; bf16 enabled
	Runtime setup	Single NVIDIA RTX 5090 GPU; maximum sequence length 1024; greedy decoding with temperature 0.0 and maximum 4 new tokens
Adapter pool	Size	252 adapters per task and base model
	Heterogeneity axes	Prompt template, LoRA rank/scale settings, target-module set, and adapter-specific sampling seed
	Grid values	Two prompt variants; $r \in \{4, 8, 12, 16, 24, 32\}$; $\alpha/r \in \{1, 1.5, 2\}$; seven target-module sets; base data seed 500
Selection	Candidate filter	Validation accuracy threshold $\tau = 0.5$ and pairwise correlation threshold $\rho = 0.4$
	Ranking and voting	Sort candidates by the chosen validation ranking score, break ties by validation accuracy, and enforce odd cardinality for majority voting

D.1 Correlation-Aware Jury Construction

Pairwise correlations are computed on error vectors. If both error vectors have zero variance and are identical, the implementation treats their correlation as 1.0 for redundancy filtering; if exactly one vector has zero variance, the pair is treated as non-blocking with correlation 0.0. This convention is conservative for duplicate no-error vectors and avoids excluding otherwise competent adapters solely because one error vector is constant.

Algorithm 1 Correlation-aware jury construction

Require: adapters $\{A_i\}_{i=1}^{N}$, selection sample set V, accuracy threshold τ, correlation threshold ρ, ranking scores r_i, optional jury size k

Ensure: selected jury $\widehat{K}$

```
Compute accuracies Acĉ_i, error vectors, and pairwise empirical phi correlations ρ̂_ij on V.
Form P_τ ← {i : Acĉ_i ≥ τ}.
Sort P_τ by increasing r_i, using accuracy as a tie-breaker.
Initialize K̂ ← ∅.
for i in the sorted order do
    if ρ̂_ij ≤ ρ for all j ∈ K̂ then
        Add i to K̂.
    end if
    if optional fixed size k is specified and |K̂| = k then
        break
    end if
end for
If odd cardinality is required and |K̂| is even, remove the lowest-ranked adapter in K̂.

return K̂
```

D.2 Main Performance Confidence Intervals

Table 5 reports the same point estimates as Table 1, with 95% paired bootstrap confidence intervals over test examples.

Table 5: Main performance with 95% bootstrap confidence intervals over test examples. Δ denotes the signed accuracy difference $\mathrm{Vote} - \mathrm{Top\text{-}1}$; positive values favor Vote. Difference intervals use paired bootstrap resampling.

		Accuracy (%)			Accuracy difference	
Model	Dataset	Vanilla	Top-1	Vote	Δ	Avg. Δ
Qwen3-0.6B	FinQA	49.49	56.38	61.22	4.85	0.75
		[44.39, 54.34]	[51.53, 61.22]	[56.38, 66.07]	[0.77, 8.93]	[-1.71, 3.22]
	TAT-QA	43.97	61.21	58.62	-2.59	
		[37.50, 50.43]	[55.17, 67.24]	[52.16, 64.67]	[-7.33, 2.16]	
	PubMedQA	16.00	61.20	61.20	0.00	
		[13.00, 19.40]	[57.00, 65.40]	[57.00, 65.40]	[-3.80, 3.80]	
Qwen3-1.7B	FinQA	48.72	65.31	67.09	1.79	1.89
		[43.62, 53.83]	[60.46, 69.90]	[62.50, 71.68]	[-2.04, 5.61]	[-0.65, 4.49]
	TAT-QA	45.69	66.81	68.10	1.29	
		[39.22, 52.16]	[60.78, 72.84]	[62.07, 74.14]	[-3.88, 6.47]	
	PubMedQA	35.60	59.60	62.20	2.60	
		[31.40, 39.80]	[55.20, 63.80]	[58.00, 66.40]	[-1.80, 7.00]	
Llama-3.2-1B	FinQA	54.08	57.40	57.40	0.00	0.78
		[49.23, 58.93]	[52.55, 62.24]	[52.30, 62.24]	[-0.77, 0.77]	[-1.68, 3.25]
	TAT-QA	53.02	63.36	62.50	-0.86	
		[46.55, 59.48]	[56.90, 69.40]	[56.47, 68.53]	[-6.90, 5.17]	
	PubMedQA	50.80	57.20	60.40	3.20	
		[46.40, 55.20]	[52.80, 61.40]	[56.20, 64.60]	[-1.00, 7.40]	
Gemma-3-1B	FinQA	50.00	52.81	53.32	0.51	3.96
		[45.15, 54.85]	[47.96, 57.65]	[48.47, 58.16]	[-4.85, 5.87]	[0.82, 7.17]
	TAT-QA	45.26	59.05	64.22	5.17	
		[38.79, 51.72]	[52.59, 65.52]	[58.19, 70.26]	[-1.29, 11.64]	
	PubMedQA	50.40	52.60	58.80	6.20	
		[46.00, 54.80]	[48.20, 57.00]	[54.40, 63.00]	[1.80, 10.60]	

D.3 Jury-Size Curve Diagnostic

Figure 3 shows the full jury-size curve for the Gemma-3-1B/TAT-QA diagnostic summarized in Table 2.

D.4 Compliance Dataset Slice-Weighted Experiments

For the compliance-dataset slice-weighted filtering experiment, we evaluated a small grid of jury sizes and target-slice weights before fixing the main-text configuration. All results in Tables 6 and 7 use the Qwen3-1.7B heterogeneous adapter pool. Specifically, we varied the jury size over $k \in \{3, 9\}$ and the target-slice weight over $w_{\mathrm{target}} \in \{1.5, 2, 3, 4\}$, while keeping the candidate pool fixed. For each target slice $\mathcal{H}_s$, slice-targeted selection used

$$w(x) = \begin{cases} w_{\mathrm{target}}, & x \in \mathcal{H}_s, \\ 1, & x \notin \mathcal{H}_s. \end{cases}$$

This weighting increases the importance of validation errors on the targeted slice while leaving all other examples at unit weight. Each weighted ranking was compared against the unweighted top-k baseline on the same candidate pool, so the only difference was whether target-slice validation errors were upweighted during ranking.

For each (k, w_{target}) pair, we report the same five test metrics used in the main text. They measure full-set error, target-slice error, pooled error on the other critical slices, error over the full critical subset, and the mass of target-slice mistakes in the overall test distribution. The last quantity is used as a simple proxy for slice-specific deployment risk.

We report $k = 9$ and $w_{\mathrm{target}} = 1.5$ as a representative moderate-weight setting, and provide the full sweep as a sensitivity analysis. The sweep is intended to show qualitative robustness rather than to tune the main reported test result. Smaller weights tended to behave more similarly to the unweighted baseline, while larger weights more aggressively specialized the jury to the target slice and could

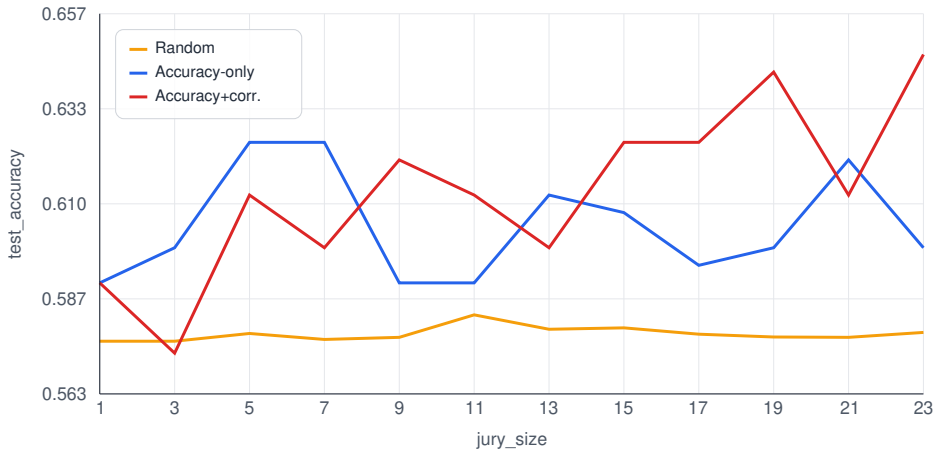


Figure 3: Jury-size curve for the Gemma-3-1B/TAT-QA diagnostic. Test accuracy is shown across odd jury sizes for random competent subsets, accuracy-only top-k selection, and the correlation-aware selection rule. The correlation-aware rule achieves the strongest full-jury result, although gains vary with jury size.

increase non-target critical error. Likewise, increasing k could improve aggregation in some settings, but could also dilute the benefit of slice targeting by admitting weaker or more correlated agents. The full sweep is therefore reported in this appendix to show that the qualitative comparison between unweighted and slice-targeted filtering is not tied to a single isolated hyperparameter choice. Table 6 provides a compact summary across (k, w_{target}) settings, while Table 7 reports the corresponding slice-level results.

D.5 Existing Assets and Licenses

Our experiments use existing public datasets, pretrained model checkpoints, and standard open-source training libraries. For the public benchmark experiments, we use the RAGBench release of TAT-QA, FinQA, and PubMedQA [6]. RAGBench is released under the CC BY 4.0 license. The original TAT-QA [24] and FinQA [2] datasets are released under CC BY 4.0, and PubMedQA [11] is released under the MIT license. We cite RAGBench and the corresponding original datasets where applicable, and we do not redistribute the original datasets.

For pretrained backbones, we use public checkpoints from the Qwen3 [21], Llama 3.1/3.2 [9, 16], Gemma 3 [8], Phi-4 [1], and Ministral 3 [17] families. Qwen3 and Ministral 3 checkpoints are released under Apache 2.0, Phi-4 checkpoints under the MIT license, Llama 3.1/3.2 checkpoints under the corresponding Meta Llama Community Licenses and acceptable use policies, and Gemma 3 checkpoints under the Gemma Terms of Use. We use these checkpoints only for research fine-tuning and evaluation, and we do not redistribute the original model weights. Any derived adapter artifacts would be subject to the corresponding base-model terms.

The implementation uses standard open-source libraries, including PyTorch [18], Hugging Face Transformers [20], and Hugging Face PEFT [14]. PyTorch is distributed under a BSD-style license, while Transformers and PEFT are distributed under Apache 2.0. We respect the licenses and terms of use of all assets used in the paper.

E Additional Comparison Results

The main performance table compares Judge CIA majority voting with the strongest validation-selected adapter from the same heterogeneous small-model pool. For completeness, Table 8 reports single-backbone comparison models, each evaluated before and after LoRA fine-tuning under the same binary-answer extraction and validation label-orientation protocol. These rows provide external context for the small-model jury results, but they are **not the central mechanism studied in this**

Table 6: Summary sensitivity sweep for compliance-dataset slice-targeted filtering. Each row corresponds to one (k, w_{target}) configuration evaluated on the same fixed candidate pool, averaged over the five target-slice runs. Target error reduction is measured relative to the corresponding unweighted top-k baseline and uses the same percentage scale as the error-rate columns. The remaining columns report average test error rates; All is the full test set, Critical is the full critical subset, and Target mass is the target-slice contribution to total test error. The selected setting is highlighted in bold.

Setting		Target error reduction	Average test error rate (%)		
k	w_{target}		All	Critical	Target mass
3	1.5	-0.20	2.84	2.11	0.60
3	2	0.64	2.84	2.26	0.44
3	3	0.45	2.84	2.11	0.48
3	4	0.24	2.88	2.41	0.52
9	**1.5**	**0.21**	**2.48**	**1.65**	**0.48**
9	2	0.00	2.52	1.80	0.52
9	3	0.02	2.44	2.11	0.52
9	4	-0.19	2.56	2.11	0.56

Table 7: Detailed slice-level results for the selected compliance-dataset setting $k = 9$ and $w_{\text{target}} = 1.5$. Each target slice corresponds to one failure-type slice within the critical subset. Target count is the number of examples for that slice in the 500-example test evaluation subset. Columns use the same shorthand as Table 3: All, Target, Other crit., Critical, and Target mass are test error rates in percent. The target-error reductions summarized in Table 6 are heterogeneous across slices; bold entries mark reductions relative to the corresponding unweighted top-k row.

			Setting		Test error rate (%)				
Target slice	Target count	Rule	k	w_{target}	All	Target	Other crit.	Critical	Target mass
Coercive inducement	100	Unweighted top-k	9	1	2.60	1.00	2.88	2.26	0.20
		Slice-weighted top-k	9	1.5	2.60	1.00	**1.92**	**1.50**	0.20
Deceptive bypass	105	Unweighted top-k	9	1	2.60	4.76	1.90	2.26	1.00
		Slice-weighted top-k	9	1.5	**2.40**	**3.81**	**0.95**	**1.50**	**0.80**
Topic evasion	95	Unweighted top-k	9	1	2.60	3.16	0.92	2.26	0.60
		Slice-weighted top-k	9	1.5	**2.20**	**2.11**	0.92	**1.50**	**0.40**
Vulnerable group	104	Unweighted top-k	9	1	2.60	1.92	2.86	2.26	0.40
		Slice-weighted top-k	9	1.5	2.60	2.88	2.86	2.26	0.60
Fact omission	96	Unweighted top-k	9	1	2.60	2.08	2.75	2.26	0.40
		Slice-weighted top-k	9	1.5	2.60	2.08	**1.83**	**1.50**	0.40

paper: they are not jury-selection or majority-vote results. For each comparison backbone, we use a LoRA configuration adapted to that model family, including the corresponding projection-module names and architecture-compatible target modules. Thus the comparison is intended to measure the effect of fine-tuning each backbone under a reasonable model-specific adapter setup, rather than forcing all models to share an identical target-module recipe.

E.1 High-Stakes Comparison Backbone Check

For completeness, Table 9 reports high-stakes compliance results for Qwen3-8B under the same validation-calibrated label-orientation protocol used for the comparison models. This check compares the vanilla and LoRA fine-tuned backbone, not a slice-weighted adapter-pool sweep.

F Routing–jury composition for catastrophic risk

This appendix is the only place where we need domain-indexed notation. This extension is not used in the main experiments, but records how the same slice-risk decomposition composes with an

Table 8: Additional comparison-model results on RAGBench. Very low vanilla scores for some instruction-tuned backbones reflect failures under the fixed binary-answer extraction protocol, not a claim about the backbone's general capability.

	Accuracy (%)					
	FinQA		TAT-QA		PubMedQA	
Comparison model	Vanilla	Fine-tuned	Vanilla	Fine-tuned	Vanilla	Fine-tuned
Gemma-3-4B	50.00	57.91	50.43	65.95	51.80	49.20
Gemma-3-12B	50.00	76.28	50.00	75.43	51.80	58.80
Llama-3.1-8B	30.87	50.00	41.38	81.47	37.80	54.00
Llama-3.2-3B	50.00	68.62	50.00	75.43	51.80	49.20
Phi-4	1.02	49.49	1.29	50.43	0.00	52.00
Phi-4-mini	50.00	50.26	50.00	50.00	51.80	52.00
Ministral-3-3B	52.04	58.16	51.29	55.60	51.20	63.80
Ministral-3-8B	50.00	61.22	50.00	55.60	51.80	54.20
Ministral-3-14B	50.77	83.16	50.43	84.05	52.00	55.60
Qwen3-4B	7.91	53.06	4.74	49.14	11.80	58.60
Qwen3-8B	50.00	52.30	50.00	50.00	51.80	51.60
Qwen3-14B	11.48	85.97	12.50	83.19	9.60	67.00

Table 9: Detailed high-stakes slice results for the Qwen3-8B comparison backbone. Each target slice corresponds to one failure-type slice in the 500-example test evaluation subset, matching the target-count convention in Table 7. All, Target, Other crit., Critical, and Target mass are test error rates in percent; Other crit. and Critical use critical-subset masks. Average rows aggregate over the five target-slice rows.

Target slice	Target count	Model	All	Target	Other crit.	Critical	Target mass
Coercive inducement	100	Qwen3-8B vanilla	41.60	39.00	45.19	44.36	7.80
		Qwen3-8B fine-tuned	**1.80**	**1.00**	**2.88**	**3.01**	**0.20**
Deceptive bypass	105	Qwen3-8B vanilla	41.60	43.81	43.81	44.36	9.20
		Qwen3-8B fine-tuned	**1.80**	**3.81**	**2.86**	**3.01**	**0.80**
Topic evasion	95	Qwen3-8B vanilla	41.60	35.79	44.95	44.36	6.80
		Qwen3-8B fine-tuned	**1.80**	**2.11**	**2.75**	**3.01**	**0.40**
Vulnerable group	104	Qwen3-8B vanilla	41.60	40.38	43.81	44.36	8.40
		Qwen3-8B fine-tuned	**1.80**	**0.00**	**3.81**	**3.01**	**0.00**
Fact omission	96	Qwen3-8B vanilla	41.60	48.96	44.04	44.36	9.40
		Qwen3-8B fine-tuned	**1.80**	**2.08**	**2.75**	**3.01**	**0.40**
Average	–	Qwen3-8B vanilla	41.60	41.59	44.36	44.36	8.32
		Qwen3-8B fine-tuned	**1.80**	**1.80**	**3.01**	**3.01**	**0.36**

upstream domain router. We combine (i) domain routing error, (ii) within-domain jury error, and (iii) catastrophic slice structure into a single decomposition.

Definition F.1 (Domain family, router, and mixed decision rule)**.** *Let $\mathfrak{D}$ be an index set of domains. For each $d \in \mathfrak{D}$, let $(\mathcal{X}_d, \mathcal{D}_d, f_d)$ denote the input space, data distribution, and ground-truth oracle for that domain. Let π be a probability distribution on $\mathfrak{D}$, so a random input is generated by first sampling $d \sim \pi$ and then sampling $x \sim \mathcal{D}_d$. A router is a possibly randomized map*

$$R : \bigsqcup_{d \in \mathfrak{D}} \mathcal{X}_d \to \mathfrak{D},$$

with predicted domain label $\hat{d} := R(x)$. For each domain d, let $\hat{y}^{(d)}$ be the decision rule of the corresponding domain jury. The mixed decision rule is

$$\hat{y}(x) := \hat{y}^{(\hat{d})}(x), \qquad \hat{d} := R(x).$$

In this appendix, $\mathrm{CatRisk}(\hat{y})$ *is evaluated under the domain mixture* $d \sim \pi$, $x \sim \mathcal{D}_d$.

Definition F.2 (True domain label and routing error on slices)**.** *Under the domain mixture, let* $d \sim \pi$ *be the true domain label and* $x \sim \mathcal{D}_d$ *the input. Given a router* R*, define* $\hat{d} := R(x)$*. For any measurable slice* $\mathcal{H}_{d,s} \subseteq \mathcal{X}_d$ *with positive probability (typically* $\mathcal{H}_{d,s} = \mathcal{H}_s \cap \mathcal{X}_d$*), define the slice routing error*

$$\alpha_{d,s} := \mathbb{P}(\hat{d} \neq d \mid x \in \mathcal{H}_{d,s}).$$

Define also the slice misroute catastrophic-error rate

$$\eta_{d,s} := \mathbb{P}\big(\mathcal{C}(x, \hat{y}(x), f_d(x)) = 1 \ \wedge\ \hat{y}(x) \neq f_d(x) \mid x \in \mathcal{H}_{d,s} \ \wedge\ \hat{d} \neq d\big) \in [0,1].$$

When the conditioning event has probability zero, we set $\eta_{d,s} := 0$.

Definition F.3 (Correct-route jury error and catastrophic amplification)**.** *For each domain* d*, let* $\hat{y}^{(d)}$ *be the domain jury decision rule (possibly after filtering). On correctly routed inputs from slice* $\mathcal{H}_{d,s}$*, define*

$$\mathrm{Err}_{d,s}(\hat{y}^{(d)}) := \mathbb{P}\big(\hat{y}^{(d)}(x) \neq f_d(x) \mid x \in \mathcal{H}_{d,s} \ \wedge\ \hat{d} = d\big),$$

and the corresponding catastrophic amplification factor

$$\kappa_{d,s} := \mathbb{P}\big(\mathcal{C}(x, \hat{y}^{(d)}(x), f_d(x)) = 1 \mid x \in \mathcal{H}_{d,s} \ \wedge\ \hat{d} = d \ \wedge\ \hat{y}^{(d)}(x) \neq f_d(x)\big) \in [0,1].$$

When the conditioning event has probability zero, we set $\kappa_{d,s} := 0$.

Proposition F.4 (Router + jury + slice composition)**.** *Let* $\hat{y}(x) := \hat{y}^{(\hat{d})}(x)$ *be the mixed decision rule. Let* $\mathcal{H} \subseteq \bigsqcup_{d \in \mathfrak{D}} \mathcal{X}_d$ *be the high-stakes set, and let* $(\mathcal{H}_{d,s})_{d \in \mathfrak{D},\, s \in \mathcal{S}}$ *be a measurable partition of* $\mathcal{H}$ *across domains and slices. Then*

$$\mathrm{CatRisk}(\hat{y}) = \sum_{d \in \mathfrak{D}} \sum_{s \in \mathcal{S}} \mathbb{P}(x \in \mathcal{H}_{d,s}) \Big[\alpha_{d,s}\, \eta_{d,s} + (1 - \alpha_{d,s})\, \kappa_{d,s}\, \mathrm{Err}_{d,s}(\hat{y}^{(d)})\Big].$$

Equivalently, using $\mathbb{P}(x \in \mathcal{H}_{d,s}) = \pi(d)\mathbb{P}(x \in \mathcal{H}_{d,s} \mid d)$*,*

$$\mathrm{CatRisk}(\hat{y}) = \sum_{d \in \mathfrak{D}} \pi(d) \sum_{s \in \mathcal{S}} \mathbb{P}(x \in \mathcal{H}_{d,s} \mid d) \Big[\alpha_{d,s}\, \eta_{d,s} + (1 - \alpha_{d,s})\, \kappa_{d,s}\, \mathrm{Err}_{d,s}(\hat{y}^{(d)})\Big].$$

Proof. For readability, write $\mathcal{C} = 1$ for $\mathcal{C}(x, \hat{y}(x), f_d(x)) = 1$ and $\hat{y} \neq f_d$ for $\hat{y}(x) \neq f_d(x)$. Decompose the catastrophic-error event on each slice $\mathcal{H}_{d,s}$ and split by whether routing is correct:

$$\mathbb{P}(\mathcal{C} = 1 \wedge \hat{y} \neq f_d \wedge x \in \mathcal{H}_{d,s}) = \mathbb{P}(x \in \mathcal{H}_{d,s})\mathbb{P}(\mathcal{C} = 1 \wedge \hat{y} \neq f_d \mid x \in \mathcal{H}_{d,s}).$$

By conditioning on the routing outcome, we have

$$\begin{aligned}\mathbb{P}(\mathcal{C} = 1 \wedge \hat{y} \neq f_d \mid x \in \mathcal{H}_{d,s}) = {} & \alpha_{d,s}\, \mathbb{P}(\mathcal{C} = 1 \wedge \hat{y} \neq f_d \mid x \in \mathcal{H}_{d,s},\, \hat{d} \neq d) \\ & + (1 - \alpha_{d,s})\, \mathbb{P}(\mathcal{C} = 1 \wedge \hat{y} \neq f_d \mid x \in \mathcal{H}_{d,s},\, \hat{d} = d).\end{aligned}$$

The misrouting term is exactly controlled by $\eta_{d,s}$: $\mathbb{P}(\mathcal{C} = 1 \wedge \hat{y} \neq f_d \mid x \in \mathcal{H}_{d,s} \wedge \hat{d} \neq d) = \eta_{d,s}$, so the first branch contributes $\alpha_{d,s}\eta_{d,s}$. On $\{\hat{d} = d\}$ we have $\hat{y}(x) = \hat{y}^{(d)}(x)$, so the chain rule gives

$$\begin{aligned}\mathbb{P}(\mathcal{C} = 1 \wedge \hat{y} \neq f_d \mid x \in \mathcal{H}_{d,s} \wedge \hat{d} = d) &= \mathbb{P}(\mathcal{C} = 1 \mid x \in \mathcal{H}_{d,s} \wedge \hat{d} = d \wedge \hat{y}^{(d)}(x) \neq f_d(x)) \\ &\quad \cdot \mathbb{P}(\hat{y}^{(d)}(x) \neq f_d(x) \mid x \in \mathcal{H}_{d,s} \wedge \hat{d} = d) \\ &= \kappa_{d,s}\, \mathrm{Err}_{d,s}(\hat{y}^{(d)}).\end{aligned}$$

Combine and sum over (d, s). □

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction describe the Judge CIA selection framework, the corresponding theoretical diagnostics, and the RAGBench and compliance evaluations; these claims are supported in Sections 3–6 and the appendices.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [No]

   Justification: The paper contains evidence from which relevant scope limits can be inferred, including non-uniform gains, validation-size dependence, diagnostic assumptions, and proxy high-stakes experiments, but it does not explicitly discuss these points as limitations.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: The theoretical statements explicitly state their assumptions in Sections 3–5; complete proofs are provided in Appendix B.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: Section 6 and Appendices D, C, and D.1 describe datasets, splits, adapter construction, training settings, decoding, selection rules, and evaluation protocols.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: The manuscript provides experimental and algorithmic details, but it does not currently include an anonymized public code/data release or supplemental reproduction commands.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

Answer: [Yes]

Justification: The experimental setup in Section 6 and Appendix D specify the splits, LoRA settings, optimizer and learning rate, decoding, pool size, and selection thresholds, with compliance-specific details in Appendix C.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: Appendix D.2 reports 95% paired bootstrap confidence intervals over test examples for the main performance table; auxiliary diagnostic tables are reported as point estimates.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.

- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [No]

Justification: Appendix D reports the GPU and runtime setup used for the experiments, but the paper does not give complete wall-clock time, memory use, or total compute for every run.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

Answer: [Yes]

Justification: The work uses public benchmark data and internally constructed compliance examples, does not involve human-subject collection, and treats high-stakes compliance experiments as diagnostic evaluations rather than deployment recommendations.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [No]

Justification: The paper does not include a dedicated broader-impact discussion because it focuses on a methodological selection diagnostic rather than proposing or deploying a user-facing system; its reliability motivation and high-stakes slice discussion are limited to the evaluation scope.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The paper does not release a new high-risk pretrained model, scraped dataset, or deployable system; the trained adapters and compliance data are used for controlled experiments in the manuscript.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: Appendix D.5 credits the existing datasets, pretrained backbones, and open-source libraries used in the experiments, and summarizes their licenses or terms of use.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [N/A]

Justification: The paper uses an internally constructed compliance dataset for evaluation but does not release it as a new asset; its construction and slice taxonomy are documented in Appendix C.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: The paper does not involve crowdsourcing experiments or research with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: No crowdsourcing or human-subject research is conducted, so IRB approval or an equivalent review is not applicable.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [No]

Justification: The research uses language-model backbones as the experimental objects, but the manuscript does not include a separate declaration of LLM usage beyond the method and experimental descriptions.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.