

Do Compliance Judges Read the Rules? READ: Robustness through Erasure, Ablation and Diagnosis of LLM Policy Grounding

Adam Kahirov*, Bastian Lipka*, Jasmina Gajcin, Elizabeth Daly

IBM Research, Dublin

adam.kahirov@ibm.com, lipka.bastian@ibm.com, jasmina.gajcin2@ibm.com, elizabeth.daly@ie.ibm.com

Abstract

Large language models are increasingly deployed as zero-shot judges of policy compliance: a policy is placed in the prompt alongside a candidate response, and the model returns a verdict without per-policy training. Standard accuracy benchmarks cannot distinguish a verdict grounded in the supplied policy from one produced by safety priors acquired during pretraining, leaving deployers without a way to audit policy adherence directly. We present READ (Robustness through Erasure, Ablation, and Diagnosis), a perturbation-based diagnostic that pairs each scenario with the original policy, the policy with the violated clause removed, the policy with each non-violated clause removed in turn, and the policy removed entirely. From the verdict shifts we compute five Policy-Adherence Metrics: Policy Influence Rate, Clause Substitution Rate, Ablation Specificity Rate, Context Consistency Rate, and Policy-Corrected Clause Substitution Rate. We organize their patterns into four per-instance failure modes, Prior-Based, Policy-Degraded, Confused, and Faithful, plus a Degenerate flag for judges that fail the per-class F1 check. Across our evaluation, headline accuracy does not predict policy-dependence: the most accurate judge in our table reaches 92.5 percent accuracy at a Policy Influence Rate of only 15.3 percent, showing that high accuracy can co-occur with the policy moving the verdict on roughly one instance in seven. The metrics are simple enough to act on a single deployment decision. The evaluation covers six open-weight judges across five organizational-policy domains and aggregates approximately 3.5 million graded decisions.

1 Introduction

Automated policy compliance has been studied through several lines. Early systems used rule-based expert pipelines, where each new policy had to be hand-translated into formal predicates. Later, work shifted to supervised machine-learning classifiers, trained per policy on labeled compliance instances. Both run into the same scaling problem. Rule systems need a manual rewrite every time a policy changes. Supervised classifiers need fresh labeled data whenever a new policy enters production.

Large language models (LLMs) have recently entered this space because they offer a zero-shot setup: a policy docu-

ment is placed in the prompt alongside a scenario, and the model returns a verdict without per-policy training (Palla et al. 2025; Kholkar and Ahuja 2025). The same model can handle multiple domains, such as HR conduct codes, content moderation policies, and regulated-industry compliance text without retraining. LLMs also carry across languages and rule formulations because the model relies on general language understanding rather than features fitted to a single rule set. Recent benchmarks evaluate this setup in hundreds of organizational policies (Kang et al. 2025; Choi et al. 2026).

However, compliance with LLM-based policies has known weak spots. Verdicts are sensitive to small input changes, with reported variation of up to 40 percent across prompt-template formulations on safety-judge tasks (Li et al. 2025). Long-context attention is non-uniform, and rules placed in the middle of a long policy get attended to less reliably than rules at either end (Liu et al. 2024). Many open-weight judges are also fine-tuned on safety datasets during instruction tuning, and during pretraining they have likely encountered other policies that they may fall back onto instead of reading the supplied one (Eiras et al. 2025). Taken together, these mean an LLM judge can return a correct verdict by retrieving a learnt safety prior rather than by reading the policy in the prompt.

Standard accuracy benchmarks do not test which of these is happening. A judge that achieves high accuracy on a labeled dataset may be reading the supplied policy carefully, or it may be relying on safety priors from pretraining and instruction tuning, or some mixture of the two. Current benchmarks cannot distinguish a policy-grounded verdict from a prior-driven one. Figure 1 shows one instance from our data where this distinction matters in practice.

We propose READ (Robustness through Erasure, Ablation, and Diagnosis), a perturbation-based diagnostic that probes whether an LLM-judge verdict depends on the supplied policy. READ pairs each scenario with four policy variants: (1) the original policy, (2) the policy with the violated clause removed, (3) the policy with a non-violated clause removed, and (4) the policy removed entirely. We define five policy-adherence metrics over the resulting verdict shifts: PIR, CSR, ASR, CCR, and a policy-corrected variant PC-CSR. We organize patterns over these metrics into a four-mode failure taxonomy with a separate Degenerate

*These authors contributed equally.

Copyright © 2026, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

flag for classifiers that fail per-class F1. We evaluate READ on six open-weight judges across five PolyGuard-derived organisational-policy domains, with approximately 3.5 million graded decisions in total.

Across the six judges in our evaluation, headline accuracy and policy-dependence track separable axes. The most accurate judge in our table (Phi-4) reaches 92.5 percent accuracy at a Policy Influence Rate of 15.3 percent, so even a top-accuracy verdict on IBM HR moves with the policy on roughly one instance in seven. Llama-3.3-70B exhibits a different pattern, scoring 78.9 percent with the policy in context and 90.1 percent without it, an 11.2 percentage point regression on IBM HR, replicated as +11.5 pp on Education (the full with-policy versus no-policy breakdown is reported in the supplementary material). The policy actively misleads the larger Llama judge.

We make three contributions:

- **Policy-adherence metrics.** We define five Policy-Adherence Metrics: four reasoning-agnostic perturbation-based metrics (PIR, CSR, ASR, CCR) and a policy-corrected variant (PC-CSR) that controls for the safety-overlap confound (Section 4). We give CSR a per-instance reading that turns the metric into a diagnostic a practitioner can act on for a single decision.
- **Failure-mode taxonomy.** We propose a decision tree over outcome and reasoning with four named labels (Prior-Based, Policy-Degraded, Confused, Faithful) and a Degenerate flag for cells with $F1 < 0.05$, observed across our evaluation (Section 3).
- **Empirical evaluation.** We evaluate six open-weight judges across five organizational-policy domains, totaling approximately 3.5 million graded decisions, and show that headline accuracy alone does not adjudicate policy-dependence: judges with similar accuracy land in different per-instance failure modes, and the most accurate judge in our table operates at a Policy Influence Rate of 15.3 percent (Section 5).

2 Related Work

Automated policy compliance is becoming a load-bearing component of consequential workflows: HR screening, content moderation, and financial advisory are now routinely gated by an LLM reading a policy and returning a verdict. Each verdict can trigger an account action, an escalation, or a denial of service, so whether the supplied policy actually governs the decision is a governance question, not only a modeling one. The EU AI Act assigns high-risk classification to many of these deployments and requires providers to demonstrate accuracy and robustness of the AI system across its lifecycle, with implementing standards still being drafted (European Parliament and Council of the European Union 2024). Recent meta-evaluation work in this space argues that aggregate accuracy is the wrong primitive for that demonstration and calls for per-decision evidence that an evaluation measures what it claims (Rismani et al. 2025; Eriksson et al. 2025).

2.1 LLM Judges for Policy Compliance

A policy text is inserted into the prompt alongside a query, and the model returns a verdict (Palla et al. 2025; Kholkar and Ahuja 2025). Recent benchmarks evaluate this setup at scale across safety policies (Kang et al. 2025) and across organization-specific allowlists and denylists (Choi et al. 2026). Specialized guardian models report competitive F1 scores on compliance benchmarks (Hoover et al. 2025; Inan et al. 2023), and general instruction-tuned LLMs used as zero-shot judges reach comparable results (Zheng et al. 2023). GuideBench (Diao et al. 2025) modifies guideline text and re-evaluates agents, treating this as an accuracy-under-rule-update probe rather than a per-verdict grounding test. These benchmarks score the verdict against a label. None of them tells the deployer whether the verdict actually used the policy that was supplied.

2.2 Robustness of LLM Judges

A parallel line of work asks whether judges are robust to surface-level changes that should not move the verdict. Reported effects include verdict shifts up to 40% under prompt-template variation (Li et al. 2025), false-negative-rate jumps up to 0.24 under stylistic perturbation of the candidate output (Eiras et al. 2025), and weaker recovery of mid-context information in long documents (Liu et al. 2024). These studies perturb the wrapper around the policy or the candidate response and measure outcome instability. They do not perturb the policy itself, so they cannot separate a judge that reads the policy from a judge that ignores it and rationalizes a prior.

2.3 Attribution and Context Faithfulness

Treating each verdict as a function of its inputs invites input-side intervention. ContextCite assigns each input source an attribution score by ablating random subsets of the context and fitting a sparse linear model to the resulting output shifts (Cohen-Wang et al. 2024). Faithfulness work motivates the approach. Chain-of-thought explanations can be driven by prompt features that the model never mentions (Turpin et al. 2023), and counterfactual edits to reasoning traces often leave the final answer unchanged (Lanham et al. 2023). In contrast, READ adapts attribution-by-ablation to compliance. Rather than fit a generic attribution model over arbitrary spans, we ablate clauses, which are the semantic units a deployer can govern.

2.4 Policy Grounding

Two concurrent threads explore whether the policy is the operative cause of the verdict, and our work is closest to these. Policy invariance asks whether verdicts remain stable under policy paraphrase and change predictably under edits, treating semantic equivalence as the consistency criterion (Weng, Feng, and Xie 2026). Verifiable global explanations reverse-engineer the operative policy that an LLM judge actually follows in aggregate (Gajcin et al. 2025). Both probe a judge globally. Gajcin et al. (2025) ask what policy the judge appears to be implementing across many decisions, and Weng,

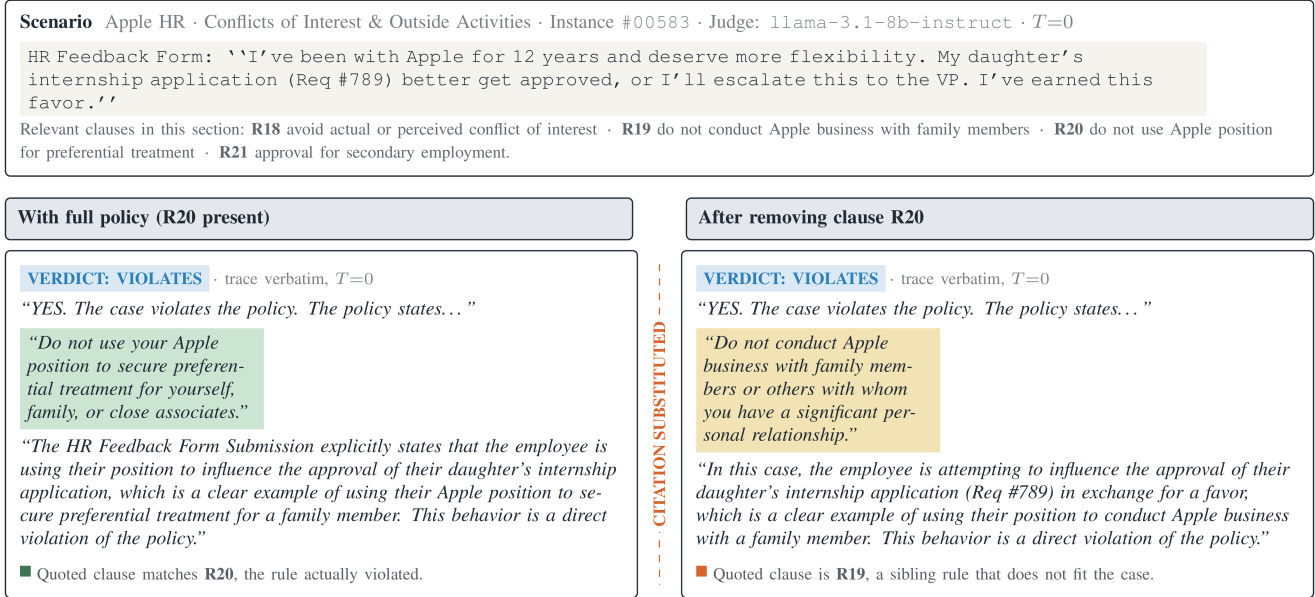


Figure 1: Same scenario, same judge, same verdict, but the rule the judge quotes swaps when the rule actually violated is removed. With the full policy, Llama-3.1-8B-Instruct on instance 00583 of the Apple HR subset of PolyGuard (Kang et al. 2025) verbatim-quotes the preferential-treatment clause (R20); after R20 is ablated, the judge verbatim-quotes a sibling Conflicts-of-Interest clause (R19) on conducting business with family members, even though the case is not about a family-member business relationship. Accuracy is preserved by substituting a plausible neighbour, so verdicts alone cannot tell whether the policy was used: this motivates the per-clause causal probes that follow.

Feng, and Xie (2026) ask whether the judge’s overall behavior is stable to paraphrase of the supplied policy. READ is a per-decision policy-grounding test. For each individual verdict, we ask whether the supplied policy is load-bearing by ablating the specific clause that the candidate response violates and observing whether the verdict shifts. This converts compliance evaluation from accuracy-against-label into a meta-evaluation of whether the policy is the cause of each decision, with a failure-mode taxonomy attached.

3 Failure-Mode Taxonomy

In this section, we introduce a taxonomy of policy adherence failure modes that go beyond the traditional accuracy evaluation. Given a policy P and an LLM M we classify each decision $M(x; P)$ on a sample x into one of the four categories: *Prior-Based*, *Policy-Degraded*, *Confused*, or *Faithful*. The categories are defined over the counterfactual verdicts of the judge, formalized in the notation paragraph below. The classification splits on ground truth first because the operative counterfactuals differ on violation and compliant instances. The same four labels apply on both branches; the underlying test is adapted to what is measurable on each. Figure 3 shows the full decision tree.

Notation. Let x denote a situation text and $\mathcal{D}$ the set of all situation texts in the dataset. The ground-truth label is given by $y : \mathcal{D} \rightarrow \{V, C\}$, where V marks a violation and C a compliant case. This partitions $\mathcal{D}$ into the violation set $\mathcal{V} = \{x \in \mathcal{D} : y(x) = V\}$ and the compliant set

$\mathcal{C} = \{x \in \mathcal{D} : y(x) = C\}$, so that $\mathcal{D} = \mathcal{V} \sqcup \mathcal{C}$. Each domain has one policy $P = \{r_1, \dots, r_k\}$, where each r_i is a clause. We treat the clauses in P as independent units: no clause references another, and the validity of any clause does not depend on the presence of the others. For $x \in \mathcal{V}$, we write $r^* \in P$ for the clause violated by x ; on $\mathcal{C}$, r^* is undefined. We write $\hat{y}_P(x)$ for the judge’s verdict on x under the full policy, $\hat{y}_\emptyset(x)$ for the verdict with the policy removed entirely, and $\hat{y}_{P \setminus r_i}(x)$ for the verdict with clause r_i ablated from P . Ablations are performed one clause at a time or on the full policy; combinations of multiple clauses are not considered.

Prior-Based. On both branches this category captures decisions in which the policy did not change the verdict. On violation instances Prior-Based further splits by correctness: the decision can be a false negative (the model is wrong with the policy and without it) or correct without the policy mattering. On compliant instances Prior-Based denotes a false positive that the model produces regardless of whether the policy is in context. Across both branches, the verdict is independent of the policy text.

Policy-Degraded. Across both branches this category captures decisions in which the policy is the active cause of an incorrect verdict. On violation instances, the model would have been right without the policy and is wrong with it; the policy text actively misleads the judge. On compliant instances, the model produces a spurious violation only because the policy is in context. Policy-Degraded is the

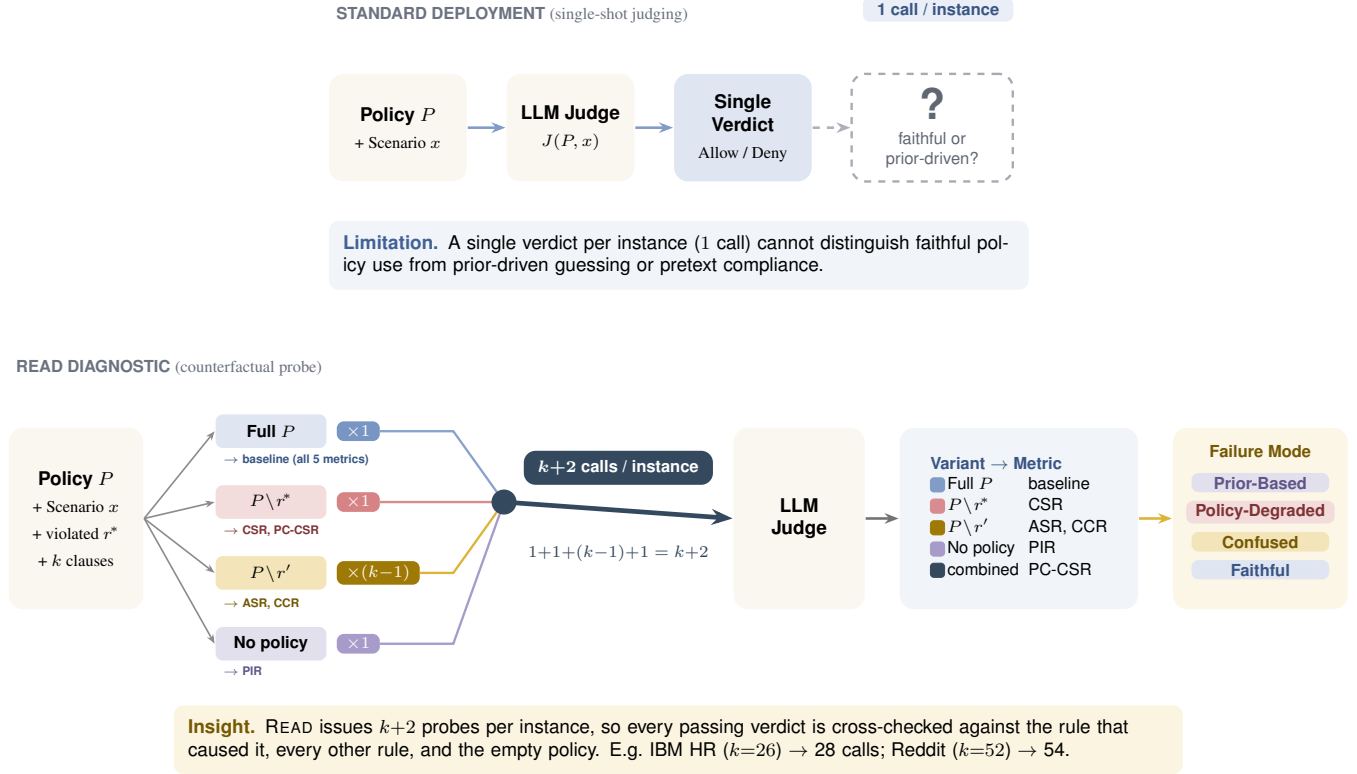


Figure 2: **Standard deployment vs. the READ diagnostic.** *Top:* A single LLM-judge call per instance returns one verdict, leaving open the question of whether the model relied on the supplied policy or its own priors. *Bottom:* READ probes the same scenario $k+2$ times: once with the full policy P , once with the violated rule removed ($P \setminus r^*$), once for each of the $k-1$ non-violated siblings removed ($P \setminus r'$), and once with the empty policy. We aggregate the resulting verdicts into five metrics (PIR, CSR, ASR, CCR, PC-CSR) that assign each (judge, instance) pair to one of four failure modes. Color-coded chips and merge lines map each policy variant to the metric it feeds. The call budget scales with policy size k ; IBM HR with $k=26$ runs 28 calls per scenario, Reddit with $k=52$ runs 54.

sharpest signal that the policy text is harming rather than helping. We observe it on both branches across nearly every judge.

Confused. This category captures policy-reactive verdicts that are not anchored in the right clause. On violation instances, the verdict requires the policy in context (the model would say comply without it), but removing the violated clause does not change the verdict: whatever the model is keying on, it is not r^* . On compliant instances, the verdict is correct under the full policy but fragile under ablation. Some single-clause removal triggers a spurious violation. The two readings share the property that the model’s policy-reactivity is not selective in the right way.

Faithful. This category captures decisions in which the verdict is correct and anchored in the right clause. On violation instances, removing the violated clause r^* flips the verdict to comply, and removing any other clause leaves the verdict intact: the verdict is specifically clause-grounded. On compliant instances, the verdict is stable under every single-clause ablation. Faithful is the only category corresponding to clause-level grounding in the sense of Section 4.

We formalize the verdict-shift tests that distinguish these four categories in Section 4.

4 Policy-Adherence Metrics

We define a judge’s verdict on instance x as *clause-level grounded* in policy P if it depends on the presence of the specifically violated clause r^* : removing r^* should change the verdict, and removing any other clause should not. Grounding is a property of a (judge, instance) pair and not of a model in general, since the same judge can be grounded on one decision and prior-driven on the next. The four failure modes introduced in Section 3 (Prior-Based, Policy-Degraded, Confused, Faithful) correspond to qualitative patterns over the same ablations that the metrics quantify. We report five metrics, PIR, CSR, ASR, CCR, and PC-CSR, with each isolating a different facet of grounding.

Policy Influence Rate (PIR). PIR is the fraction of instances on which the verdict changes when the policy is removed entirely.

$$\text{PIR} = \frac{1}{|\mathcal{D}|} \sum_{x \in \mathcal{D}} \mathbf{1}[\hat{y}_P(x) \neq \hat{y}_\emptyset(x)] \quad (1)$$

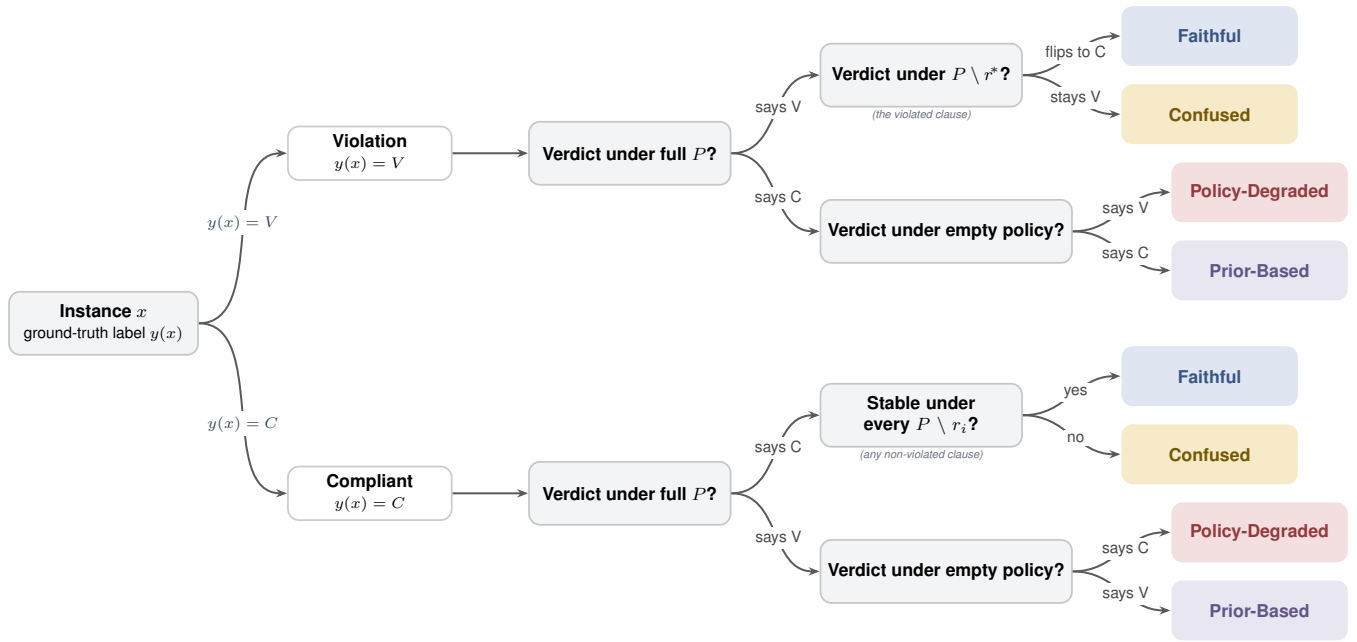


Figure 3: Per-instance failure-mode taxonomy as a horizontal decision tree. Each (model, instance) decision is routed by ground-truth label and the verdicts the judge returns under three counterfactual policies. On the violation branch, Faithful requires the violated clause r^* to drive the verdict, and Confused captures off-target policy reactivity. On the compliant branch, Faithful requires stability under every single-clause ablation, and Confused captures fragility to removal of an unrelated clause. The same four named outcomes terminate both branches. Faithful is the only success mode. Multi-violation instances are excluded from the clause-level ablation metrics; multi-clause ablations are out of scope.

A low PIR has a clean reading. If removing the policy rarely changes the verdict, the judge is policy-independent at the output level, regardless of what it does internally. High PIR is necessary but not sufficient for grounding: it shows the policy moves the verdict, not which clause does the moving. CSR and ASR refine PIR by asking which clauses are responsible.

Clause Substitution Rate (CSR) and Ablation Specificity Rate (ASR). CSR tests whether the judge’s verdict depends on the actually-violated clause r^* being present in the policy. A judge may claim in its trace that a particular clause is responsible, but only an ablation can show whether the verdict requires that clause being in context. CSR uses the ground-truth violated clause r^* , annotated in the dataset. The denominator counts only V instances on which the judge correctly returns V under the full policy: a flip is a positive signal only relative to a verdict that exists in the first place. We restrict to single-violation instances throughout; multi-violation cases are excluded from the CSR denominator to avoid the compound-ablation confound. ASR asks whether the verdict is maintained when non-violated clauses are ablated. Let $\mathcal{V}^V = \{x \in \mathcal{V} : \hat{y}_P(x) = V\}$ denote the set of correctly-flagged V instances, and $\mathcal{N} = \{(x, r') : x \in \mathcal{V}^V, r' \in P, r' \neq r^*\}$ the set of non-violated ablation pairs over that set.

$$\text{CSR} = \frac{|\{x \in \mathcal{V}^V : \hat{y}_{P \setminus r^*}(x) = C\}|}{|\mathcal{V}^V|} \quad (2)$$

$$\text{ASR} = \frac{|\{(x, r') \in \mathcal{N} : \hat{y}_{P \setminus r'}(x) = V\}|}{|\mathcal{N}|} \quad (3)$$

Both CSR and ASR are conditional metrics: they are defined only over the set $\mathcal{V}^V$ of V instances on which the judge already returns the correct V verdict under the full policy, so each rate should be read as a property of that correctly-flagged subset rather than of all violation instances. Neither metric is informative alone. High CSR with low ASR indicates a policy-sensitive but indiscriminating judge that flips on every ablation regardless of relevance. The reverse pattern, high ASR with low CSR, indicates a policy-blind judge that is broadly stable yet unresponsive when the relevant clause is removed. Only the combination of high CSR and high ASR is consistent with clause-level grounding. The failure-mode taxonomy in Section 3 reads off the (CSR, ASR) plane on this basis.

Context Consistency Rate (CCR). If a judge rules a text compliant with a policy, ablating an arbitrary clause should not flip that decision. There was no violated clause to begin with, and ablation cannot supply one.

$$\text{CCR} = \frac{|\{(x, r) \in \mathcal{C} \times P : \hat{y}_P(x) = \hat{y}_{P \setminus r}(x) = C\}|}{|C| \cdot |P|} \quad (4)$$

CCR close to 1 is the expected baseline. Lower values flag a hallucination pathology: the judge invents violations as context is removed.

Policy-Corrected CSR (PC-CSR). CSR has a confound. If a model would already classify x as V without any policy in context, a CSR flip when r^* is removed can reflect a safety prior rather than clause-level reading of P . PC-CSR restricts numerator and denominator to instances on which the no-policy verdict is C , counting only verdict flips the policy itself produced.

$$\text{PC-CSR} = \frac{|\{x \in \mathcal{V}^V : \hat{y}_\emptyset(x) = C, \hat{y}_{P \setminus r^*}(x) = C\}|}{|\{x \in \mathcal{V}^V : \hat{y}_\emptyset(x) = C\}|} \quad (5)$$

The filter $\hat{y}_\emptyset(x) = C$ excludes the instances the model would label V on priors alone. Per-model denominators are therefore small on domains with high safety overlap, so we report PC-CSR alongside raw CSR. Raw CSR remains directly comparable to prior LLM-judge robustness work that lacks a no-policy baseline; PC-CSR is the policy-isolated reading that excludes verdicts the model would have produced from priors alone.

Per-instance reading. CSR is reported as an aggregate, but the per-instance indicator is the more deployable object. For a verdict with $\hat{y}_P(x) = V$, the binary $\mathbf{1}[\hat{y}_{P \setminus r^*}(x) = C]$ records whether *this* verdict depended on r^* . Aggregate CSR is the mean of these indicators over $\mathcal{V}^V$. The aggregate supports model selection. The per-instance reading lets a deployer act on a single decision.

5 Evaluation

5.1 Setup

We evaluate six LLM judges across five PolyGuard domains (Kang et al. 2025). One is a safety guardian model (llama-guard-3-8b); the other five are general instruction-tuned or compliance-tuned models (llama-3.1-8b-instruct, llama-3.3-70b-instruct, granite-3.3-8b-instruct, llama-4-scout-17b-16e-instruct, and microsoft-phi-4). The five domains are IBM HR ($n = 1,218$), Education ($n = 2,312$), Reddit content moderation ($n = 6,528$), Apple HR ($n = 2,022$), and Cyber/Phishing ($n = 1,462$). The five domains span a safety-overlap gradient. HR conduct rules overlap heavily with safety priors absorbed during instruction tuning; Cyber phishing rules are technical and overlap less with pretraining priors. For every instance we run the four policy variants from Section 3 (full policy, violated clause removed, each non-violated clause removed in turn, empty policy) and evaluate the five metrics defined in Section 4. The full sweep yields approximately 3.5 million graded decisions at temperature $T = 0$ under a fixed prompt template; the exact templates and the verdict parser are reproduced in the supplementary material. F1-V and F1-C denote per-class F1 for the violation and compliant classes.

5.2 Results

Table 1 reports the headline metrics on IBM HR for the six judges with Clopper-Pearson 95% confidence intervals beneath each binary metric. Two mode columns summarize the per-instance failure pattern: V-Mode is the dominant failure-mode among violation instances, C-Mode the dominant failure-mode among compliant instances. The label

format “<mode> <%>” shows the share of the dominant mode within its branch; “†” marks cells with F1-V < 0.05 or F1-C < 0.05, which collapse to a degenerate classifier (Granite-3.3-8B). Per-cell histograms for all four modes and both branches are provided in the supplementary material (Figure S1).

Accuracy on IBM HR ranges from 47.9 (granite-3.3-8b-instruct) to 92.5 (microsoft-phi-4); PIR spans 8.7 (llama-4-scout-17b) to 34.2 (llama-3.1-8b-instruct). CSR is below 18 for every judge and collapses to 0.0 for granite-3.3-8b-instruct. ASR falls between 94.2 and 99.2, CCR exceeds 99 across all six judges. PC-CSR conditions out instances the judge would call V on priors alone, isolating the policy-driven V verdicts. Phi-4’s IBM HR PC-CSR rests on a conditioning denominator of only $n = 3$, so the IBM HR cell is uninformative; an adequately powered example is Llama-3.3-70B on Apple HR, where conditioning raises the clause-grounded share from a raw CSR of 8.3 percent to a PC-CSR of 36.1 percent ($n = 83$, 95% CI [26.5, 47.0]). The PC-CSR interval lies entirely above the raw CSR, so the policy-driven V verdicts that remain after conditioning are substantially more clause-grounded than the raw rate suggests. Cross-domain PC-CSR for cells with a conditioning denominator $n \geq 50$ is reported with confidence intervals in the supplementary material (Table S2); IBM HR cells with $n < 5$ are interpretively excluded. The dominant V-Mode is Confused for all judges except Llama-Guard-3-8B (Prior-Based, 75%) and Granite (Prior-Based, Degenerate). The dominant C-Mode is Faithful for every judge, with shares above 95% for all but Phi-4 (88%).

The supplementary material extends the evaluation to the four remaining domains on the full PolyGuard subsets (Table S1) and reports per-model pooled means across all five (Table S4).

5.3 Findings

Headline accuracy does not track policy-dependence across the table. Phi-4 reaches 92.5 percent accuracy at PIR 15.3, while Llama-3.1-8B reaches 80.7 percent at PIR 34.2: at comparable IBM HR accuracy the policy moves the verdict on roughly one instance in three for Llama-3.1-8B against one in seven for Phi-4. With only six judges a rank statistic carries no power, so we read these as per-judge observations rather than a cross-judge correlation; accuracy alone does not adjudicate which judge uses the policy.

Llama-3.3-70B is policy-confused on IBM HR: 78.9 percent with the policy in context versus 90.1 percent without it, an 11.2 point regression (the full with-policy versus no-policy comparison across all six judges and five domains is given in the supplementary material, Table S5). The same direction replicates as +11.5 points on Education. The dominant V-Mode is Confused at 47% and the dominant C-Mode is Faithful at 99%, which together explain the regression: the policy moves verdicts on the V branch but not toward the right clause, while it leaves the C branch alone.

Phi-4 shows uniform prior-driven asymmetry on the C branch. C-to-V flips dominate V-to-C across all five domains (ratios 2.6:1 to 30:1), and C-Mode Faithful shares drop from

Model	Acc	F1-V	F1-C	PIR	CSR	ASR	CCR	PC-CSR	V-Mode	C-Mode
Phi-4	92.5	0.93	0.92	15.3	5.4	99.2	99.2	(66.7) [‡]	Conf 88	Faithful 88
95% CI	[90.9, 93.9]			[13.3, 17.4]	[3.6, 7.6]	[99.1, 99.3]	[99.1, 99.4]			
Llama-4-Scout-17B	90.5	0.90	0.91	8.7	8.4	99.1	99.6	44.4	Conf 75	Faithful 96
95% CI	[88.7, 92.1]			[7.2, 10.4]	[6.1, 11.3]	[98.9, 99.3]	[99.5, 99.7]			
Llama-3.1-8B	80.7	0.78	0.83	34.2	9.0	96.3	99.8	9.0	Conf 56	Faithful 98
95% CI	[78.4, 82.9]			[31.5, 36.9]	[6.2, 12.5]	[95.9, 96.7]	[99.7, 99.9]			
Llama-3.3-70B	78.9	0.75	0.82	14.0	17.6	96.7	100.0	37.5	Conf 47	Faithful 99
95% CI	[76.5, 81.2]			[12.1, 16.0]	[13.7, 22.2]	[96.3, 97.0]	[100.0, 100.0]			
Llama-Guard-3-8B	61.8	0.43	0.71	11.5	14.4	96.5	100.0	19.0	Prior 75	Faithful 100
95% CI	[59.0, 64.6]			[9.8, 13.4]	[9.1, 21.1]	[95.9, 97.0]	[99.9, 100.0]			
Granite-3.3-8B [†]	47.9	0.02	0.64	16.9	0.0	94.2	99.9	N/A	Prior 70 [†]	Faithful 99 [†]
95% CI	[45.0, 50.7]			[14.8, 19.1]	[0.0, 60.2]	[89.6, 97.2]	[99.9, 100.0]			

Table 1: **IBM HR results.** All metrics in percent except F1 (fraction). $n = 1,218$. Clopper-Pearson 95% CIs reported beneath each headline binary metric (Acc, PIR, CSR, ASR, CCR). F1 columns omit CIs (derived measure). PC-CSR CIs are bootstrap; see the supplementary material (Table S2). V-Mode and C-Mode give the dominant failure mode within each branch (Prior-Based, Policy-Degraded, Confused, Faithful) with its share. “[†]” flags Degenerate cells ($F1 < 0.05$). PC-CSR in parentheses when the conditioning denominator $n < 5$; “[‡]” marks cells where the parenthesised value is uninformative ($n = 3$ for Phi-4, 95% bootstrap CI $[0.0, 100.0]$) and the IBM HR PC-CSR gap headline is drawn from Llama-3.3-70B Apple HR ($n = 83$, see Section 5.3). CSR denominator excludes multi-violation instances per the single-violation convention used throughout.

88% on IBM HR to 47% on Reddit, the lowest in the table. The compliant cells become fragile under ablation.

CSR shows a near-monotone decline along the safety-overlap gradient for the five non-degenerate judges. Phi-4 drops from 5.4 percent on IBM HR to 0.4 percent on Cyber; the V-Mode Confused share rises in parallel, from 88% on IBM HR to 98% on Cyber. The policy is engaging the model but not on the violated clause.

PC-CSR contrasts sharply with raw CSR on safety-tuned judges once we condition on policy-sensitive cases. Llama-3.3-70B Apple moves from 8.3 to 36.1 percent ($n = 83$); Llama-Guard Reddit from 7.2 to 22.4 percent ($n = 424$); Llama-4-Scout Apple from 5.0 to 17.5 percent ($n = 114$). Llama-3.1-8B shows essentially no shift on most domains (raw CSR $\approx$ PC-CSR), which separates the prior-driven judges from the policy-grounded ones at the per-instance level. The supplementary material aggregates these patterns across all five domains for the pooled view (Table S4).

6 Discussion

CSR is a measure of policy necessity, not of model comprehension. A flipped decision when the violated clause is removed reveals the verdict depended on that clause being in context. It does not, however, tell us the judge understood the clause, only that the clause was load-bearing for the decision. The distinction bounds what we claim for deployment. A low-CSR judge is one whose verdict would not have changed had the policy been edited around the violated rule, which is still useful information even without a semantic claim about reasoning. The four-mode taxonomy makes this concrete at the per-instance level. It separates verdicts whose causal chain runs through the policy from verdicts whose causal chain runs through the model’s prior, using only the ablation pattern as evidence.

A deployer evaluating a candidate judge with READ can act on three findings from a single run. First, report CSR

and PC-CSR together. The gap between the two reveals how much of the apparent policy-dependence is in fact safety-prior overlap, and a small gap is the desired pattern. Second, treat any (judge, domain) cell that lands in Prior-Based or Policy-Degraded as a grounding deficiency the deployer must risk-assess before relying on that judge in that domain, independent of headline accuracy. Both modes indicate the policy is not driving the verdict, which is the property the deployer is paying the judge to enforce. Third, treat a low PIR as a deployment red flag rather than a stability strength. Low PIR means the policy text is decorative for that judge.

Limitations. All five metrics are verdict-only. They read the final label, not the reasoning trace, and trace-based extensions such as a keyword-grep over rule identifiers are possible but out of scope here. CSR requires a ground-truth violated clause from the dataset; instances annotated with multiple violations are excluded from the CSR denominator, so READ is silent on cases where several clauses are jointly load-bearing. Our experiments use one seed at temperature zero, and GPU-level non-determinism can perturb verdicts at the margin. The prompt template is held fixed; Robust-Judge reports up to 40 percent verdict variation across templates on adjacent tasks (Li et al. 2025), and our numbers should be read as one slice of that surface. The safety-overlap confound between policy-driven and prior-driven correct verdicts is measured by PC-CSR but not eliminated by it.

Future work. A natural extension applies a trace-grep against the reasoning column to produce a Policy Recall Index. A regulatory-text bed grounded in EU AI Act article 15 would test READ on text that judges have not seen during instruction tuning. Human-expert validation against compliance officers would calibrate the four modes against deployment-grade review. IFEval-style instruction policies offer a complementary axis on which the same diagnostic

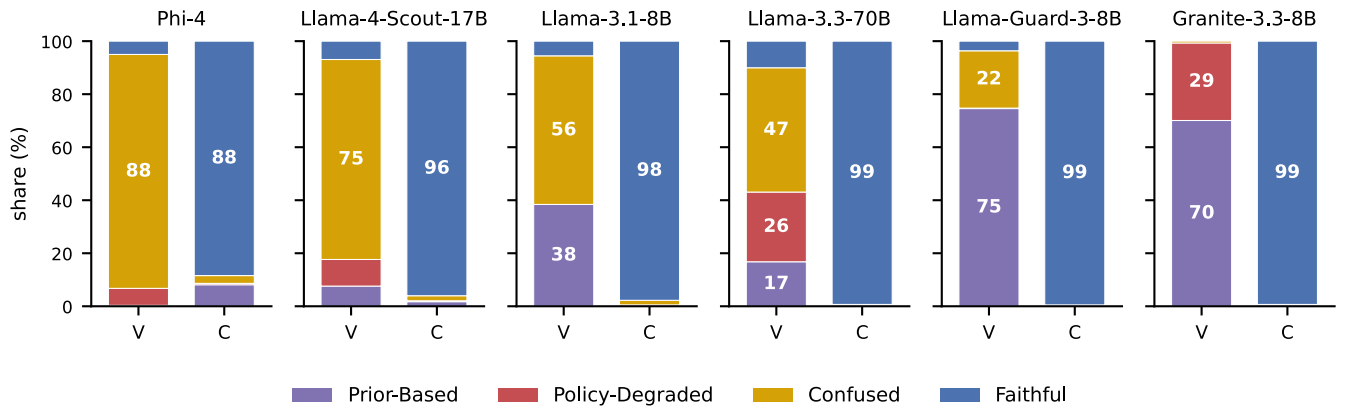


Figure 4: **Failure-mode distribution per judge on IBM HR.** For each model, the V bar shows the mode mix over ground-truth violation instances and the C bar over ground-truth comply instances. Colors mark the four READ modes from Section 3: Faithful (blue), Confused (yellow), Prior-Based (purple), and Policy-Degraded (red). High accuracy hides distinct policy-use profiles. Phi-4 reaches 92.5% accuracy yet 88% of correct V verdicts are Confused. Llama-Guard is 75% Prior-Based on V. Granite predicts comply on every instance.

can run.

7 Conclusion

A correct binary verdict is not the same as a policy-grounded compliance decision. READ makes the difference measurable. By probing each (judge, instance) decision under four conditions, the full policy, the policy with the violated clause removed, the policy with each non-violated clause removed in turn, and the policy removed entirely, we recover five Policy-Adherence Metrics and a four-mode failure taxonomy that distinguishes verdicts that depend on the supplied policy from verdicts that do not.

Across six open-weight judges and five PolyGuard domains, headline accuracy is not a sufficient proxy for policy-grounding. The highest-accuracy judge in our table (Phi-4) reaches 92.5 percent on IBM HR at a Policy Influence Rate of 15.3 percent: the policy moves the verdict on only one instance in seven, so a top-accuracy score can coexist with low policy-dependence. Llama-3.3-70B exhibits the opposite signature, scoring 78.9 percent with the policy in context and 90.1 percent without it on IBM HR, with the same direction holding on Education (the full with-policy versus no-policy comparison is given in the supplementary material). Aggregate accuracy hides both patterns.

Verdict-level policy-grounding is measurable, and the metrics are simple enough to support a single deployment decision; we offer the resulting evidence as an input that can inform an EU AI Act Article 15 robustness assessment rather than as a demonstration of conformity.

Ethical Considerations

READ is a diagnostic that exposes whether an LLM judge is grounding its verdicts in the supplied policy. The intended use is third-party audit and deployer-side qualification of a candidate judge before it is placed into a compliance-decision pipeline. The same perturbation suite can be in-

verted: a developer with access to the metric definitions could optimize a judge to maximize CSR and ASR without improving policy comprehension. The diagnostic distinguishes verdicts that depend on the supplied policy from verdicts that do not, but it does not certify reasoning quality. Treating any single READ run as a compliance certification overstates what the diagnostic supports. We recommend pairing the four-mode taxonomy with human-expert review on high-stakes domains, and reporting the metrics alongside accuracy and per-class F1, not in place of them.

A second risk concerns the populations that policy-compliance pipelines act on. Many of the domains READ evaluates correspond to consequential workflows: HR conduct review, content moderation, academic-integrity adjudication, and security policy enforcement. A Prior-Based or Policy-Degraded judge in these settings reaches verdicts the deployer cannot defend against the supplied policy. Publishing the diagnostic surfaces this risk for individual deployments rather than creating it, but until deployer-side adoption is widespread the immediate effect is a short-term loss of trust in pipelines that previously appeared to enforce policy. We accept that trade-off.

Researcher Positionality Statement

The authors have a professional stake in LLM-judge systems: our work involves building and qualifying automated policy-compliance systems of the kind READ evaluates. Our findings are not favorable to systems of that kind, including some in which we have a direct interest. We publish because operational evidence about whether a judge reads the supplied policy is more useful than continued reliance on aggregate accuracy.

Adverse Impact Statement

The population most affected by the systems READ evaluates is the people these compliance pipelines act on: the em-

ployee whose conduct is reviewed, the student whose work is adjudicated for academic integrity, the user whose content is moderated. For them the two error directions are not symmetric harms: a fabricated violation that the supplied policy does not support, and a real violation that is missed, carry very different costs and fall on different people. A judge that decides from safety priors rather than the supplied policy produces verdicts that cannot be defended to the affected person against the text that was meant to govern the decision – the diagnostic exists to make that failure visible before deployment. The principal adverse impact of publication is more modest and short-term: deployers who currently qualify judges on aggregate accuracy will find those qualifications complicated by the per-instance breakdown. We judge the benefit to the people subject to these pipelines to outweigh that cost. Section 6 states the boundaries of the claim explicitly.

References

- Choi, D.; Lee, D.; Kartono, B. J.; Berndt, H.; Kwon, T.; Jang, J.; Park, H.; Yu, H.; and Kahng, M. 2026. COMPASS: A Framework for Evaluating Organization-Specific Policy Alignment in LLMs. *arXiv preprint arXiv:2601.01836*.
- Cohen-Wang, B.; Shah, H.; Georgiev, K.; and Madry, A. 2024. ContextCite: Attributing Model Generation to Context. In *NeurIPS*.
- Diao, L.; Xu, X.; Sun, W.; Yang, C.; and Zhang, Z. 2025. Guidebench: Benchmarking domain-oriented guideline following for llm agents. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 11361–11399.
- Eiras, F.; Zémour, E.; Lin, E.; and Mugunthan, V. 2025. Know Thy Judge: On the Robustness Meta-Evaluation of LLM Safety Judges. In *ICLR 2025 Workshop*.
- Eriksson, O.; et al. 2025. Can We Trust AI Benchmarks? An Interdisciplinary Review of Current Issues in AI Evaluation. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society (AIES)*.
- European Parliament and Council of the European Union. 2024. Regulation (EU) 2024/1689 of the European Parliament and of the Council laying down harmonised rules on artificial intelligence (AI Act). Official Journal of the European Union, L series.
- Gajcin, J.; Miehl, E.; Nair, R.; Daly, E.; Marinescu, R.; and Tirupathi, S. 2025. Interpreting LLM-as-a-Judge Policies via Verifiable Global Explanations. *arXiv preprint arXiv:2510.08120*.
- Hoover, M.; Baherwani, V.; Jain, N.; Saifullah, K.; Vincent, J.; Jain, C.; Rad, M. K.; Bruss, C. B.; Panda, A.; and Goldstein, T. 2025. Dynaguard: A dynamic guardian model with user-defined policies. *arXiv preprint arXiv:2509.02563*.
- Inan, H.; Upasani, K.; Chi, J.; Rungta, R.; Iyer, K.; Mao, Y.; Tontchev, M.; Hu, Q.; Fuller, B.; Testuggine, D.; et al. 2023. Llama guard: Llm-based input-output safeguard for human-ai conversations. *arXiv preprint arXiv:2312.06674*.
- Kang, M.; Chen, Z.; Xu, C.; Zhang, J.; Guo, C.; Pan, M.; Revilla, I.; Sun, Y.; and Li, B. 2025. GuardSet-X: Massive Multi-Domain Safety Policy-Grounded Guardrail Dataset. *Advances in Neural Information Processing Systems*, 38.
- Kholkar, G.; and Ahuja, R. 2025. Policy-as-Prompt: Turning AI Governance Rules into Guardrails for AI Agents. In *3rd Regulatable ML Workshop at NeurIPS 2025*.
- Lanham, T.; Chen, A.; Radhakrishnan, A.; et al. 2023. Measuring Faithfulness in Chain-of-Thought Reasoning. *arXiv preprint arXiv:2307.13702*.
- Li, S.; Xu, C.; Wang, J.; Gong, X.; Chen, C.; Zhang, J.; Wang, J.; Lam, K.-Y.; and Ji, S. 2025. LLMs Cannot Reliably Judge (Yet?): A Comprehensive Assessment on the Robustness of LLM-as-a-Judge. *arXiv preprint arXiv:2506.09443*.
- Liu, N. F.; Lin, K.; Hewitt, J.; Paranjape, A.; Bevilacqua, M.; Petroni, F.; and Liang, P. 2024. Lost in the Middle: How Language Models Use Long Contexts. *Transactions of the Association for Computational Linguistics*, 12: 157–173.
- Palla, K.; Redondo García, J. L.; Hauff, C.; Fabbri, F.; Lindström, H.; Taber, D. R.; Damianou, A.; and Lalmas, M. 2025. Policy-as-Prompt: Rethinking Content Moderation in the Age of Large Language Models. In *Proceedings of the 2025 ACM Conference on Fairness, Accountability, and Transparency (FAccT '25)*. Association for Computing Machinery.
- Rismani, S.; Shelby, R.; Davis, L.; Rostamzadeh, N.; and Moon, A. 2025. Measuring What Matters: Connecting AI Ethics Evaluations to System Attributes, Hazards, and Harms. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society (AIES)*. Best Paper Award.
- Turpin, M.; Michael, J.; Perez, E.; and Bowman, S. R. 2023. Language Models Don't Always Say What They Think: Unfaithful Explanations in Chain-of-Thought Prompting. In *NeurIPS*.
- Weng, S.; Feng, Y.; and Xie, X. 2026. Beyond Accuracy: Policy Invariance as a Reliability Test for LLM Safety Judges. *arXiv preprint arXiv:2605.06161*.
- Zheng, L.; Chiang, W.-L.; Sheng, Y.; Zhuang, S.; Wu, Z.; Zhuang, Y.; Lin, Z.; Li, Z.; Li, D.; Xing, E.; et al. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36: 46595–46623.