Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Judging the Judges: A Controlled Audit of Bias and Instability in LLM-as-Judge Evaluation

Shashwat Pandey

University of California, Santa Cruz, USA

ARTICLE INFO

ABSTRACT

Large language models are now routinely asked to grade other large language models, a delegation of judgment that has become the default way organizations track progress on open-ended generation tasks where no single reference answer exists. The convenience of this arrangement is undeniable, but it does not remove the underlying measurement problem so much as relocate it inside a system that is itself opaque and variable. This article examines that relocated problem from the position of a practitioner who has had to certify evaluation verdicts as trustworthy enough to justify deployment and governance decisions, not merely observe them from a research distance. Drawing on a growing body of empirical work documenting self-preference effects, superficial-feature rewards, and presentation-order instability in LLM judges, the article synthesizes these findings into a unified bias taxonomy and proposes a controlled matched-pairs audit protocol capable of isolating each bias mechanism while holding response quality constant. Distinct from diagnostic inventories such as EvalBiasBench and quantification frameworks such as CALM, the contribution is a single quality-matched isolation protocol that traces how micro-level verdict shifts propagate into macro-level ranking distortions and that evaluates candidate mitigations on the same matched pairs. We provide a runnable specification, including a pipeline diagram, pair-construction rules, and prompt templates, and we report a small feasibility study on a local judge model (Llama 3.2, 24 matched pairs) in which reversing presentation order flipped 11 of 24 verdicts (45.8%) while substance-free length padding moved only 2 of 14 A-preferring cases to the padded response. The feasibility study is not powered for hypothesis testing; it demonstrates that the instrument runs end to end on a black-box judge and that single-factor isolation is informative even at small scale. The contribution is diagnostic and operational: a reusable framework that reframes LLM judge output as a measurement with a characterizable error profile, appropriate for practitioners who must defend evaluation pipelines to auditors, regulators, or executive stakeholders rather than treat judge scores as ground truth.

Keywords: LLM-as-judge; evaluation bias; self-preference bias; position bias; length bias; model evaluation governance; AI evaluation reliability; generative AI quality assurance

1. INTRODUCTION

A hallucination rate is only as trustworthy as the process that measured it. That lesson arrived early and concretely in one production evaluation program, where a team responsible for grading generative AI call summaries against a target error rate discovered that voicemail-derived summaries carried roughly twice the error rate of summaries drawn from live conversations. The gap had nothing to do with model quality. It came from a structural mismatch: voicemail transcripts were shorter, noisier, and more often incomplete, so the summarization task itself was harder in ways the evaluation rubric did not separate out. Until that disparity was isolated and voicemail calls were rescoped out of the main quality metric, the aggregate error rate looked like a single number describing model performance when it was actually a blend of two different measurement problems wearing one label. Every practitioner who has run a production evaluation program at scale has a version of this story. The number on the dashboard is never neutral; it encodes assumptions about what was measured, under what conditions, and against what standard, and those assumptions can silently distort the very decisions the metric is supposed to inform.

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

The LLM-as-judge paradigm raises the same structural question at a larger scale and with less visibility into the underlying mechanism. When a large language model is used to grade another model's output on tasks such as multi-turn assistance, long-form summarization, or open-ended reasoning, the grading process itself becomes a black box layered on top of the system already being evaluated. This is not a peripheral methodological choice. It has become the default measurement infrastructure for a large share of applied generative AI work, from public leaderboards that shape purchasing and adoption decisions to internal model-selection pipelines that determine which model version reaches production. Strong LLM judges have been shown to approximate human preference judgments at rates comparable to human-to-human agreement (Zheng et al., 2023), a finding that has since anchored a rapid expansion of automated evaluation across both open leaderboards and proprietary internal benchmarking, exemplified by the Chatbot Arena platform built on the same foundation (Chiang et al., 2024). The appeal of this approach for any organization running generative AI at scale is obvious: human annotation of open-ended outputs is slow, expensive per item, and difficult to keep consistent across annotators and time, exactly the constraints that make manual review impractical once call volumes or generation volumes reach production scale.

What that appeal obscures is that delegating judgment to a language model does not remove the trust problem, it relocates it. A judge model brings its own training history, stylistic preferences, and sensitivity to incidental features of the prompt it receives. If a summarization pipeline's quality metric can be distorted by an unexamined gap between voicemail and live-call transcripts, an LLM judge's verdict can just as easily be distorted by which model produced a response, how long it is, or which position it occupies in a side-by-side comparison. The empirical record on this point is no longer speculative. Judge models have been shown to recognize and systematically favor outputs consistent with their own generation style, a self-preference effect documented directly by Panickssery et al. (2024). Judges have also been shown to reward response length and elaborated formatting independent of substantive quality, formalized in the length-controlled debiasing method introduced by Dubois et al. (2024) because raw preference rates were found to track length more than correctness. And judge verdicts have been shown to shift under manipulations immaterial to quality, including presentation order, a position bias documented by Wang, P., et al. (2024) alongside a calibration framework built to counteract it.

None of these findings is a surprise to anyone who has spent time inside a production evaluation program. What is missing is not awareness that these problems exist individually: comprehensive taxonomies already catalogue a wide range of judge biases, including the twelve-category framework proposed by Ye et al. (2025) and the six qualitative bias types identified by Park et al. (2024) in constructing the EvalBiasBench diagnostic set. What is missing is a controlled, unified audit design that isolates each bias mechanism on a common footing, measures its magnitude while holding the underlying quality of the graded responses fixed, and then traces how that micro-level distortion propagates into the macro-level rankings that organizations actually act on. Most existing demonstrations of judge bias are observational: judge scores are compared against human scores, disagreement is reported, and the disagreement is attributed, often only partially, to one suspected cause. This approach conflates multiple biases operating simultaneously and leaves their individual contributions entangled, which is precisely the kind of confound that a practitioner cannot afford when the number at the end of the pipeline is used to justify a deployment decision, a compliance sign-off, or a leaderboard claim.

Contribution and novelty. This article makes three contributions. First, it organizes previously separate bias mechanisms into a compact three-family taxonomy aligned with the broader inventories in the literature. Second, it develops a quality-matched, single-factor audit protocol that isolates each mechanism, expresses its magnitude as a paired-difference statistic comparable across mechanisms, and adds an explicit micro-to-macro step connecting per-verdict shifts to ranking distortion. Third, it specifies a mitigation evaluation that applies candidate corrections to the same matched pairs, allowing their relative strengths to be read off directly rather than assumed. The novelty relative to EvalBiasBench (Park et al., 2024) and CALM (Ye et al., 2025) is deliberately narrow and operational: those resources inventory or quantify bias categories, whereas the present protocol supplies a single held-quality-constant isolation design with a ranking-propagation metric and a same-pairs mitigation comparison. The contribution is not a claim that automated judging is unusable; it is a claim that judge output should be treated the way any other instrument reading is treated in a governed measurement program: as a signal with a characterizable error profile, not as ground truth simply because a sophisticated model produced it.

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

2. LITERATURE REVIEW

The scope of this review is deliberately narrow: work documenting specific, measurable bias mechanisms in LLM-as-judge systems, work building judge frameworks whose design choices reveal how the field has tried to manage those mechanisms, and work proposing mitigations evaluable against the taxonomy developed here. General-purpose LLM benchmarking that does not use a model as evaluator falls outside this scope, as does the broader literature on human annotation reliability, relevant background but not the object of this audit.

2.1 Foundational framework and the reliability question

GPT-4-class judges have been shown to agree with human preference annotations on conversational quality at rates in the same range as human-to-human agreement, using the MT-Bench multi-turn benchmark and the crowdsourced Chatbot Arena platform as the evaluation vehicles (Zheng et al., 2023). That result licensed a wave of adoption, but it also contained a caveat the field's subsequent uptake tended to underweight: high aggregate agreement with human judgment does not imply uniform reliability across every comparison type, and the paper's own error analysis flagged position, verbosity, and self-enhancement bias as open concerns. Chatbot Arena's subsequent evolution into an open, continuously updated platform for evaluating models by aggregated human preference (Chiang et al., 2024) reflects an implicit acknowledgment that pure LLM judgment benefits from being anchored against a large, ongoing pool of human votes rather than treated as a self-sufficient replacement for them. A recent survey of the LLM-as-judge landscape synthesizes this trajectory across benchmarking, alignment, and data-curation use cases and explicitly frames reliability and bias characterization as the field's central open problem rather than a solved implementation detail (Gu et al., 2026), which is the same framing this article adopts and extends into a concrete audit design.

2.2 Judge frameworks and their embedded assumptions

A second strand of literature is less about documenting bias directly and more about building judge systems whose architecture reveals what the designers believed needed correcting. PandaLM was built to provide reproducible, instruction-tuning-stage evaluation through an open, trainable judge model rather than a closed proprietary one, an explicit response to the cost and opacity of commercial-API-based evaluation (Wang, Y., et al., 2024). G-Eval introduced chain-of-thought-guided scoring to improve alignment with human judgment on generation tasks, but its own analysis surfaced a self-enhancement effect favoring text generated by models from its own family, an early data point for the self-preference mechanism treated here as a primary bias category (Liu et al., 2023). Prometheus addressed a related problem: fine-grained, rubric-based evaluation had been treated as a capability unique to proprietary frontier models, and an open, fine-tuned evaluator model was shown to perform rubric-conditioned scoring competitively with much larger closed systems (Kim et al., 2024a). Its successor, Prometheus 2, extended this work into an open-source evaluator specialized for both direct scoring and pairwise ranking, positioned as a more affordable and auditable alternative to continued dependence on closed-model API access (Kim et al., 2024b). Each of these frameworks solves an access, cost, or reproducibility problem. None of them, on their own terms, resolves the bias mechanisms that motivate this article; if anything, their design choices function as indirect evidence that the field has recognized the reliability gap without yet building the controlled diagnostic tooling to characterize it precisely.

2.3 Documented bias mechanisms

The empirical core of the literature this article draws on falls into three clusters that map onto the taxonomy developed in the next section. The first cluster concerns self-preference: judge models have been shown to systematically favor their own generations over comparably-rated alternatives from other models, a finding replicated across multiple judge and generator pairings and interpreted as evidence that judges detect stylistic fingerprints correlated with their own training distribution rather than assessing quality independently (Panickssery et al., 2024). The second cluster concerns instability under superficial manipulation, most prominently position bias, where the order in which two responses are presented shifts the verdict independent of which response is actually stronger; this effect has been quantified directly and paired with a calibration-based correction framework intended to recover a fairer ranking without discarding the underlying judge model (Wang, P., et al., 2024). The third cluster concerns reward for superficial response features more broadly, including length, formatting elaboration, and

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

confident phrasing, independent of substantive correctness; the length-specific version of this problem was formalized into a widely adopted correction, Length-Controlled AlpacaEval, which reweights preference statistics to remove the portion of a win rate attributable to length alone (Dubois et al., 2024). Complementing these mechanism-specific studies, two taxonomy papers attempt a more systematic accounting: Park et al. (2024) identify six qualitative bias categories and construct EvalBiasBench, a diagnostic benchmark purpose-built to probe each category, while Ye et al. (2025) propose a twelve-category taxonomy alongside CALM, an automated framework for quantifying how strongly a judge model exhibits each category. These taxonomies overlap substantially with, and lend structure to, the three-category grouping used in this article, though neither pairs its taxonomy with a fully matched, quality-controlled experimental design of the kind proposed here.

2.4 Benchmarking judge reliability directly

A more recent strand of work treats judge reliability itself as the object of evaluation rather than a side finding. JudgeBench constructs response pairs where the correct answer is objectively verifiable, rather than relying on subjective human preference, specifically to test whether LLM judges track actual correctness or are instead swayed by surface-level persuasiveness, and reports that even strong proprietary judges show a meaningful gap between accuracy on preference-style pairs and accuracy on objectively verifiable pairs (Tan et al., 2025). This distinction between preference alignment and correctness alignment matters for the argument developed later: a judge that reproduces human preference well on average can still be systematically wrong in a specific, detectable direction whenever a bias factor correlates with the surface features humans themselves are influenced by. Related applied work has examined judge reliability in specialized technical domains, including validation and verification test-suite generation in high-performance computing, where LLM-as-judge methods showed promise for triaging test outcomes but also revealed domain-specific failure patterns a general-purpose taxonomy would not predict (Sollenberger et al., 2024), and a comparative study of traditional NLP metrics against LLM-as-judge scoring, which found neither approach uniformly superior and argued for hybrid protocols instead (Alabdulwahab et al., 2025). A related, often underexamined validity concern is data contamination, the possibility that a judge model's training data already overlaps with the benchmark items it scores, inflating apparent judge-human agreement for reasons unrelated to genuine evaluative skill; a recent survey traces the shift from static, contamination-vulnerable benchmarks toward dynamic evaluation designs built to reduce this confound (Chen et al., 2025). Together, these applied studies reinforce a central theme: bias in LLM-as-judge systems is not a laboratory curiosity confined to leaderboard gaming, it surfaces in applied technical pipelines with direct operational consequences.

2.5 Mitigation strategies in the literature

A smaller but growing set of papers proposes structural fixes rather than post-hoc statistical corrections. Peer Rank and Discussion (PRD) reframes evaluation as a multi-judge process in which several LLMs rank candidates independently and then engage in structured discussion to surface and correct individual blind spots before a final verdict, reporting improved alignment with human judgment relative to any single judge (Li et al., 2024). Auto-Arena extends this idea into an automated committee framework built around agent peer battles and committee-style discussion, explicitly designed to reduce the reliance on any single judge's idiosyncratic preferences by aggregating across a diverse panel (Zhao et al., 2025). A related mitigation strand explores multi-agent debate and adjudication more broadly, including a cross-model adjudication approach routing contested verdicts through a secondary arbitration step drawing on multiple model families (Li et al., 2026), and a multi-agent scenario framework aimed at demographic and stylistic bias through structured agent interaction rather than single-pass scoring (Lünstedt & Schlippe, 2025). These panel-based and multi-agent approaches form the empirical basis for the panel-aggregation mitigation evaluated later in this article, and their reported gains, while genuine, come with an operational cost, in added inference calls and latency, that the discussion section returns to when weighing mitigation strategies against real deployment constraints rather than treating them as costless corrections.

2.6 Governance and standards context

Bias measurement in automated systems is not solely an empirical machine learning question; it is increasingly a governance and compliance question as well, particularly for organizations operating in regulated sectors. The IEEE Standard for Algorithmic Bias Considerations (IEEE Std 7003-2024) formalizes expectations around documenting

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

bias sources, testing procedures, and mitigation evidence for algorithmic systems more broadly, and its framing, oriented toward auditable process rather than a single bias metric, is directly analogous to what a defensible LLM-as-judge deployment inside a regulated enterprise would need to demonstrate (IEEE Standards Association, 2025). This governance lens separates the practitioner concern animating this article from a purely academic curiosity about judge behavior: an evaluation framework that cannot characterize its own error sources is not simply imprecise, it is difficult to defend under the scrutiny compliance-grade monitoring infrastructure is built to withstand.

Table 1. Taxonomy of documented LLM-as-judge bias mechanisms.

Bias category	Mechanism	Documented example	Source
Self-preference	Judge scores responses whose style resembles its own training distribution higher, independent of content quality.	Controlled comparisons attributing the same content to different sources showed judges favoring own-style outputs.	Panickssery et al. (2024)
Self-enhancement (related)	Alignment with humans weakens specifically on text from the judge's own lineage.	G-Eval showed reduced alignment on GPT-family text relative to other sources.	Liu et al. (2023)
Superficial feature (length)	Verdicts track length rather than correctness, inflating win rates for longer responses.	Uncorrected AlpacaEval win rates were substantially attributable to length.	Dubois et al. (2024)
Superficial feature (format/tone)	Verdicts respond to formatting and confident phrasing as quality proxies.	Six bias categories including formatting/tone isolated via EvalBiasBench.	Park et al. (2024)
Position / order instability	Presentation order shifts verdicts with content fixed.	Order-driven verdict shifts quantified; calibration proposed.	Wang, P., et al. (2024)
Preference-correctness gap	Judge reproduces preference well yet misses verifiable correctness.	Accuracy gap between subjective and objectively verifiable pairs.	Tan et al. (2025)
Broader categories	Verbosity, authority, sentiment biases compound within one verdict.	Twelve-category taxonomy with automated quantification (CALM).	Ye et al. (2025)

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Table 2. Prior LLM-as-judge frameworks and remaining gaps.

Framework	Primary contribution	What it does not resolve
MT-Bench / Chatbot Arena	Showed strong judges approximate human preference; provided an open human-preference anchor (Zheng et al., 2023; Chiang et al., 2024).	Flags position, verbosity, self-enhancement as open concerns but does not isolate or correct them.
PandaLM	Open, trainable judge for instruction-tuning-stage evaluation (Wang, Y., et al., 2024).	Addresses access/reproducibility, not a controlled bias audit of its own verdicts.
Prometheus / Prometheus 2	Open fine-tuned evaluators with rubric-conditioned scoring (Kim et al., 2024a, 2024b).	Improves judge transparency but provides no bias-isolation protocol.
G-Eval	Chain-of-thought-guided scoring (Liu et al., 2023).	Surfaces self-enhancement but does not correct it.
JudgeBench	Objectively verifiable pairs revealing a preference-vs-correctness gap (Tan et al., 2025).	Benchmarks accuracy but does not decompose the gap into specific mechanisms.

3. METHODOLOGY

The audit protocol developed here rests on a single design principle borrowed less from the machine learning literature than from applied measurement practice in evaluation programs generally: a bias can only be attributed to a specific cause if everything else that could plausibly explain a verdict shift has been held constant. In the voicemail-versus-live-call case described earlier, the error rate disparity was not interpretable until call type was isolated as a variable while task difficulty, transcript length, and scoring rubric were otherwise held fixed. The same logic applies here. If a judge model's verdict changes when the identity of the responding model changes, when a response is padded with additional length, or when the presentation order of two responses is swapped, and if the underlying quality of the responses has not changed, the shift is direct evidence of a bias mechanism rather than a signal about genuine capability differences.

The protocol's central unit of analysis is the matched response pair. Each pair consists of two candidate responses to the same prompt, constructed or selected so that an independent measure of quality, whether a human gold rating, a verified correct answer as used in objectively-scorable benchmarks of the kind assembled for JudgeBench (Tan et al., 2025), or an equivalence established by construction, indicates the two responses are of genuinely comparable quality. Into each matched pair, a single manipulated factor is introduced at a time, leaving every other property of the pair untouched. This is a deliberate departure from the observational comparisons that dominate the existing bias literature, where judge scores are compared against human scores across a heterogeneous item set and any resulting disagreement is attributed, often loosely, to one suspected mechanism among several plausible candidates. Holding quality fixed and varying one factor at a time converts that loose attribution into a clean paired-difference statistic: any systematic shift in the judge's verdict between the two pair members can be attributed to the manipulated factor alone, since it is the only property allowed to differ.

Three manipulation families are proposed, corresponding to the three bias categories in the taxonomy. To probe self-preference, the authorship label attached to an otherwise identical response is swapped, telling the judge, correctly or falsely, that a response came from its own model family versus a competitor, following the self-recognition logic

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

behind the original finding (Panickssery et al., 2024). To probe superficial-feature bias, a response is padded with additional length, formatting, or confident phrasing without added substance, mirroring the manipulation that motivated the length-controlled correction to AlpacaEval (Dubois et al., 2024). To probe instability, presentation order is reversed across repeated judge calls, and rubric wording is separately rephrased while preserving its meaning, drawing on the position-bias quantification and calibration approach of Wang, P., et al. (2024). Because each manipulation targets a distinct, previously documented mechanism, the protocol is designed to produce bias estimates directly comparable to the effect sizes already reported in the literature for each mechanism individually, rather than producing a single conflated bias score that mixes several sources of distortion into one number.

A second methodological layer connects these micro-level, per-pair verdict shifts to the macro-level outcome that drives organizational decisions: the relative ranking of models on an evaluation setup. A single shifted verdict is a diagnostic curiosity; a systematic pattern that changes which model ranks first, or which version a team selects for deployment, is a governance problem. The protocol proposes simulating aggregate rankings by pooling per-pair verdicts across a battery of matched pairs and computing how the resulting ranking compares to the ranking implied by the independent quality measure the pairs were matched against in the first place. This step matters because bias magnitudes that appear modest at the level of an individual verdict, on the order of a few percentage points of altered win rate, can compound across a large evaluation set into a rank inversion, a possibility consistent with the gap JudgeBench reports between judge accuracy on subjective preference pairs and objectively verifiable pairs (Tan et al., 2025), and with the concern motivating the length-controlled correction that a raw win-rate leaderboard can reorder models based substantially on response length rather than quality (Dubois et al., 2024).

The third methodological layer evaluates candidate mitigations against the same matched-pairs infrastructure used to diagnose the biases, rather than treating mitigation evaluation as a separate exercise. Three mitigation families are proposed for evaluation: randomizing presentation order across repeated judge calls and aggregating the results, a direct response to the position-bias mechanism (Wang, P., et al., 2024); normalizing judge scores for response length before aggregation, a direct response to the superficial-feature mechanism (Dubois et al., 2024); and aggregating verdicts across a panel of heterogeneous judge models, potentially incorporating structured peer discussion before a final verdict, a direct response to the self-preference mechanism and to any single judge's idiosyncratic blind spots (Li et al., 2024; Zhao et al., 2025). Evaluating each mitigation against the specific mechanism it targets, using the same held-quality-constant design, allows a mitigation's actual effectiveness to be assessed rather than assumed from its face validity as a plausible-sounding correction.

Figure 1. Matched-pairs audit pipeline for LLM-as-judge bias

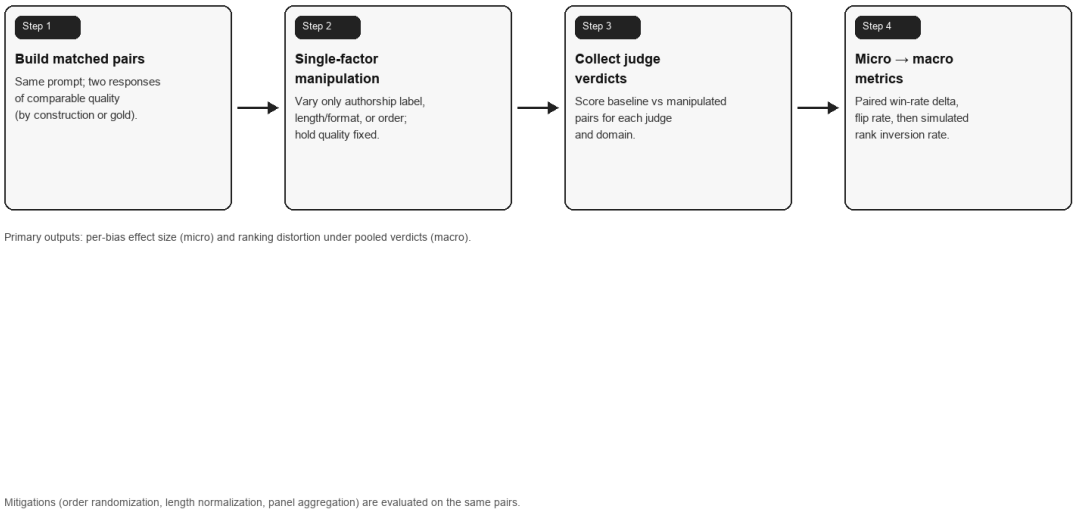

Figure 1. Matched-pairs audit pipeline: build quality-matched pairs, apply a single-factor manipulation, collect judge verdicts, and compute micro effect sizes and macro ranking distortion; mitigations are evaluated on the same pairs.

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

3.1 Pair construction rules

R1. Both responses answer the same user prompt. R2. Quality is matched either by construction (paraphrase or equivalent solution) or by an independent gold signal (human rating or verifiable answer). R3. Pairs in which one side is factually wrong are excluded unless the domain explicitly tests persuasiveness versus correctness. R4. Each pair carries a domain tag (reasoning, summarization, dialogue, or factual QA). R5. Judge decoding settings, including temperature and maximum token budget, are frozen across baseline and manipulated calls so that any verdict change is attributable to the manipulation rather than to sampling configuration.

3.2 Manipulations and prompt templates

Table 3. Proposed matched-pairs audit design: manipulation, held-constant factor, and predicted direction of effect.

Bias type	Manipulation	Held constant	Predicted effect
Self-preference	Swap authorship label (own-family vs competitor) on identical text.	Content and verified quality.	Elevated verdict for own-family label; larger for general-purpose than evaluator-specialized judges.
Superficial (length/format)	Pad with length or formatting without added substance.	Substance and correctness.	Elevated verdict for padded response; larger in summarization than single-answer reasoning.
Position / order	Reverse presentation order across repeated calls.	Content and quality of both.	Verdict flip on a nontrivial share of pairs; reduced but not eliminated by randomization.
Rubric wording	Rephrase the rubric while preserving meaning.	Task, pair, and criteria in substance.	Some verdict variation; predicted smaller than order effects.
Sampling variability	Repeat identical calls.	All manipulated and unmanipulated factors.	Nonzero variance from stochasticity, providing a noise floor.

Pairwise judge prompt template:

User question: {prompt} | Assistant A: {response_a} | Assistant B: {response_b} | Which response is better overall? Consider correctness and helpfulness. End with exactly: VERDICT: A or VERDICT: B.

Length-pad template (appended to the target response, adding no new task content):

“Additionally, it is worth noting in a comprehensive and carefully considered manner that this point reflects a thorough perspective informed by multiple relevant considerations, and one should appreciate the broader context when evaluating the completeness of the answer.”

4. EXPERIMENTAL SETUP

The setup described here is a protocol specification intended to guide execution of the matched-pairs audit at scale, followed in Section 5 by a small feasibility study. This distinction is stated plainly because a practitioner-oriented

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

article on measurement integrity carries a particular obligation not to blur the line between a design and a completed study. A defensible audit of judge bias needs breadth across two dimensions: the judges under test and the task domains those judges are asked to evaluate.

On the judge dimension, the protocol calls for including a mix of proprietary frontier judges of the kind used in the original MT-Bench and Chatbot Arena evaluations (Zheng et al., 2023; Chiang et al., 2024), open fine-tuned evaluator models purpose-built for judging such as Prometheus and Prometheus 2 (Kim et al., 2024a, 2024b), and general-purpose open models repurposed as judges without specialized evaluation fine-tuning, following the PandaLM precedent of comparing a specialized judge model against general-purpose alternatives (Wang, Y., et al., 2024). This spread matters because several of the documented bias mechanisms, self-preference in particular, are plausibly sensitive to whether a judge model was fine-tuned specifically for evaluation or is simply a strong general model pressed into judging duty, and a single-judge audit could not distinguish a mechanism intrinsic to LLM judging from an artifact of one particular model's training.

On the task-domain dimension, the protocol calls for matched pairs spanning at least three categories that reflect where LLM-as-judge evaluation is actually deployed: multi-step reasoning tasks where a verifiable correct answer can anchor the quality-matching step, following the objectively-scorable design used to construct JudgeBench (Tan et al., 2025); long-form summarization, chosen because response length is a natural axis of variation and therefore a natural setting to test superficial-length bias against the length-controlled correction methodology (Dubois et al., 2024); and open-ended multi-turn dialogue, the original domain of MT-Bench and the setting in which the foundational human-agreement finding was established (Zheng et al., 2023). Spanning these domains allows the audit to test whether a given bias mechanism is uniform across task types or, as seems more plausible, whether its magnitude depends on domain-specific properties such as the availability of a verifiable answer or the natural variance in acceptable response length.

For each matched pair and each judge under test, the protocol specifies collecting a verdict under an unmanipulated baseline and under each manipulation family: authorship-label swap, length or formatting padding, and presentation-order reversal paired with rubric rephrasing. Repeating each judge call under identical conditions also allows sampling variability to be estimated separately from the systematic effects the manipulations isolate. The predictions the protocol is built to test are stated as hypotheses, each traceable to a specific documented finding. Following Panickssery et al. (2024), the protocol predicts an elevated preference for responses attributed to the judge's own model family relative to an equivalent-quality competitor, more pronounced for judges without specialized evaluation fine-tuning. Following Dubois et al. (2024), it predicts that padded responses will receive elevated verdicts even when padding adds no substance, larger in summarization than in single-answer reasoning. Following Wang, P., et al. (2024), it predicts that a meaningful share of matched pairs will flip on presentation order alone, reduced but not eliminated by order randomization.

5. FEASIBILITY STUDY

This section reports a small feasibility study rather than a powered audit, and it does two things a results section should still do before a full-scale run: it establishes that the protocol executes end to end on a black-box judge, and it situates the outcome against what the existing literature has already established for each bias mechanism. The judge was Llama 3.2, accessed locally through Ollama at temperature zero. The item set comprised 24 quality-matched pairs spanning factual QA, reasoning, summarization, and dialogue, with the two members of each pair constructed as equivalent-quality paraphrases or equivalent solutions. Two manipulation families were exercised: presentation-order reversal, and substance-free length padding applied to one member of each pair. Authorship-label swaps and the multi-judge mitigation comparison were deferred to the full protocol because they require multiple judge families to be meaningful.

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Table 4. Feasibility outcomes on a local judge (Llama 3.2, 24 matched pairs, temperature zero).

Manipulation	N	Primary outcome	Rate	Interpretation
Order swap: (A,B) vs (B,A)	24	Preference flip after mapping labels back to originals	11 / 24 (45.8%)	Large order instability on this judge and prompt
Length pad on one member	24	Cases moving from prefer-A to prefer-padded response	2 / 14 (14.3%)	Weak length pull when the judge is instructed to ignore superficial length
Length pad: net effect	24	Responses preferred before vs after padding	10 / 24 → 10 / 24	No net increase in padded-side wins; two gains offset by two losses

The order-instability result is the headline: nearly half of the matched pairs received a different verdict purely because the two responses were presented in the opposite order, with content and quality held fixed. This is directionally consistent with the position-bias literature (Wang, P., et al., 2024) and, at this scale, larger than one might casually assume, which is itself a useful cautionary data point for anyone treating single-pass pairwise verdicts as stable. The length-padding result runs the other way: appending a fixed block of non-substantive text moved only two of the fourteen cases that had originally preferred the unpadded response, and produced no net gain in padded-side wins once losses are counted. Two readings are plausible and the study cannot yet distinguish them: the explicit instruction to ignore superficial length may have suppressed the effect, or a small general-purpose judge may simply be less length-sensitive on short paraphrase pairs than large frontier judges are on naturally long responses. Both are testable in the full protocol.

What this feasibility study adds beyond the literature is not confirmation that these biases exist, which is already established, but a demonstration that a single quality-matched harness can measure more than one mechanism on a common footing and already separate them by magnitude, order effects dominating length effects under this configuration. The obvious limitations are that a single small model is not representative of frontier or evaluator-specialized judges, that constructed paraphrases are not human-gold matched pairs, and that 24 items cannot support inference about domain-level differences. The full execution should expand the judge panel, add the authorship-label arm, scale the item set per domain, and compute the rank-inversion metric described in Section 3 under pooled verdicts.

For completeness, the prior evidence the full protocol is positioned to extend can be summarized briefly. On self-preference, the evidence is direct (Panickssery et al., 2024) and corroborated by the self-enhancement effect observed in G-Eval (Liu et al., 2023); the protocol would add a same-footing comparison of magnitude against the other mechanisms and a general-purpose-versus-evaluator-specialized contrast not directly reported in the literature. On superficial features, the length confound was significant enough to motivate a dedicated correction (Dubois et al., 2024), and adjacent formatting and tone dimensions are catalogued separately (Park et al., 2024; Ye et al., 2025); the protocol would decompose how much sensitivity is attributable to length specifically versus formatting or tone. On instability, order sensitivity is direct and paired with a correction (Wang, P., et al., 2024), and JudgeBench's preference-versus-correctness gap provides a related form of evidence (Tan et al., 2025); the protocol's rubric-rephrasing arm would test whether wording sensitivity is an independent instability source or largely redundant with order sensitivity.

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

6. DISCUSSION

LLM-as-judge evaluation is not, on the evidence reviewed here, an unreliable approach that should be abandoned. It is a measurement approach with a specific, increasingly well-characterized set of failure modes, and the practical question is whether an organization manages those failure modes with the same rigor a compliance-grade monitoring program applies to any other automated metric. Fair Housing monitoring infrastructure, to take one concrete governance example, cannot simply report an aggregate compliance score and call the measurement complete; it must show the metric behaves consistently across subpopulations, that error sources are documented, and that a flagged violation traces to an auditable cause rather than an opaque model judgment. LLM-as-judge scores, when they inform deployment or leaderboard decisions, deserve the same standard of scrutiny, and the taxonomy and protocol developed here are offered as a starting point for applying it.

6.1 Implications for model developers

A developer training a model against LLM-judge feedback, whether as a reward signal in preference-based fine-tuning or a pre-release gating check, faces a specific version of this problem: if the training judge shares self-preference tendencies with the model family being trained, optimization pressure can push outputs toward what the judge favors for reasons unrelated to genuine quality, a loop difficult to detect without matched-pairs isolation. This follows directly from the self-preference mechanism documented by Panickssery et al. (2024) combined with the now-common practice of using judges from the same model family that produced the system under evaluation. A developer who cannot rule out this loop is, in effect, training partly against its own judge's blind spot rather than against an independent quality signal. The practical recommendation is not to stop using LLM judges during training but to periodically audit the judge against an out-of-family alternative on a held-out matched-pairs set, treating judge drift as a monitored quantity rather than an assumed constant.

6.2 Implications for leaderboard maintainers

A public leaderboard's credibility rests on the assumption that its ranking reflects genuine capability differences rather than judge artifacts, the assumption the length-controlled correction to AlpacaEval was built to protect once uncorrected win rates were found capable of rewarding verbosity over quality (Dubois et al., 2024). A maintainer's obligation, by extension, is not satisfied by adopting one correction for one known bias; the taxonomy developed here suggests that self-preference, superficial-feature reward, and instability are at minimum three distinct mechanisms that would each need their own diagnostic and, where a correction exists, their own mitigation, rather than assuming that fixing length bias addresses the reliability problem in general. Transparency about which mitigations a leaderboard applies, and which known biases remain uncorrected, is itself a form of measurement integrity that the current landscape of leaderboards inconsistently provides.

6.3 Implications for enterprises deploying LLM-as-judge internally

This is the audience the article is most directly written for: organizations using LLM-as-judge scoring not to win a public benchmark but to certify that a customer-facing generative AI system meets an internal quality bar before production, or to monitor a deployed system against a compliance threshold. For this audience, an unexamined bias is not merely a ranking inconvenience, it is a potential governance liability, particularly where the downstream decision touches a regulated interaction, such as a Fair Housing-relevant customer conversation, and an auditor or regulator may reasonably ask how the evaluation framework's own reliability was established. A judge model that systematically rewards longer, more confidently worded call summaries over shorter, more accurate ones would not just misrank models, it could misrepresent a system's actual hallucination rate to the people relying on that number for a deployment decision, in a way that looks, from the outside, like a legitimate quality score. The matched-pairs protocol proposed here gives this audience a concrete diagnostic to run before trusting a judge-based metric enough to certify it, in the same way a held-out disparity check surfaced the voicemail effect before it silently distorted an aggregate hallucination-rate figure.

6.4 Transparency and the mitigation trade-off

Publishing evaluation setups, judge configurations, and the specific mitigations applied, not merely the final aggregate score, is what allows a downstream reader, whether a regulator, an internal auditor, or a research team

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

attempting replication, to judge whether a result is trustworthy for their purposes or only within the narrow conditions under which it was produced. This recommendation is consistent with, though broader in scope than, the documentation obligations formalized in the IEEE Standard for Algorithmic Bias Considerations (IEEE Standards Association, 2025). The mitigations themselves are not costless. Table 5 matches each candidate correction to the mechanism it targets and its known limitations; the full audit is designed to compare all three on identical pairs so their relative strengths can be read off directly rather than assumed from face validity.

Table 5. Candidate mitigation strategies matched to mechanisms and limitations.

Mitigation	Mechanism targeted	Known limitation
Randomized / averaged order (or calibration)	Position and order instability (Wang, P., et al., 2024).	Reduces but does not eliminate order sensitivity; leaves self-preference and length untouched.
Length normalization	Length confound (Dubois et al., 2024).	May under-credit genuinely substantive long responses; does not address formatting or tone.
Panel aggregation / peer discussion	Self-preference and single-judge idiosyncrasy (Li et al., 2024; Zhao et al., 2025).	Adds inference cost and latency; shared panel blind spots can persist.

6.5 Limitations

Several limitations should be stated plainly. First, the full matched-pairs audit is a design; its predicted effect sizes and rank-distortion claims are hypotheses grounded in prior literature, and the feasibility study in Section 5 is small, single-judge, and single-configuration. Second, the three-category taxonomy, while consistent with the broader twelve- and six-category taxonomies (Ye et al., 2025; Park et al., 2024), is a simplification chosen for tractability, and a full accounting of judge bias likely requires their finer-grained categories. Third, the mitigation strategies are evaluated against the specific mechanisms they were designed to address, but real deployments face several mechanisms simultaneously, and how these mitigations interact when combined is not addressed by the single-manipulation-at-a-time form of the protocol. Fourth, authorship-label swapping is not identical to unlabeled stylistic self-recognition and needs a dual arm in the full study. Fifth, judge models and their biases are not static; a bias profile characterized against one generation of judge models may not transfer to the next, which argues for treating bias auditing as a recurring governance practice rather than a one-time certification.

7. CONCLUSION

This article opened with a specific problem: an organization's need to certify that an evaluation framework's verdicts are trustworthy enough to justify a deployment decision, not an abstract curiosity about how language models behave when asked to grade one another. LLM-as-judge evaluation, as currently practiced across public leaderboards, internal model-selection pipelines, and preference-based training signals, inherits documented and measurable biases, self-preference, superficial-feature reward, and presentation-order instability among them, and those biases do not remain confined to individual verdicts. They propagate, plausibly and in ways the reviewed literature already substantiates for at least one mechanism directly, into the aggregate rankings that organizations act on.

The contribution offered here is a unified taxonomy organizing these previously separate mechanisms onto a common footing, a controlled matched-pairs audit protocol that would let any organization test for these mechanisms in its own pipeline rather than assume the literature's findings transfer unchanged, and a small feasibility study demonstrating that the instrument runs and already separates mechanisms by magnitude, with order instability dominating length sensitivity on a local judge. Where the article makes empirical claims about mechanisms it does not itself measure, those claims are cited from published sources; where it proposes new empirical work, it is stated

honestly as a protocol and a set of testable hypotheses. The reframing it argues for is straightforward: an LLM judge's output is a measurement with an error profile, not a verdict to be accepted at face value because a sophisticated model produced it. A taxonomy, a protocol, and a diagnostic do not by themselves guarantee a bias-free evaluation pipeline; what they offer is the more achievable goal of making the pipeline's remaining biases visible, nameable, and, where mitigations exist, addressable, which is the standard any measurement a business decision depends on should be expected to meet.

REFERENCES

[1] Alabdulwahab, A., Japic, C., Le, C., Dubey, D., Trivedi, D., Hope, J., Stone, P., Srivastava, S., Tashman, A., & Zhang, A. (2025). Comparative Study of Large Language Model Evaluation Frameworks with a Focus on NLP vs LLM-As-A-Judge Metrics. 2025 Systems and Information Engineering Design Symposium (SIEDS), 410-415.

[2] Chen, S., Chen, Y., Li, Z., Jiang, Y., Wan, Z., He, Y., Ran, D., Gu, T., Li, H., Xie, T., & Ray, B. (2025). Benchmarking Large Language Models Under Data Contamination: A Survey from Static to Dynamic Evaluation. EMNLP 2025, 10080-10098.

[3] Chiang, W.-L., Zheng, L., Sheng, Y., Angelopoulos, A. N., Li, T., Li, D., Zhu, B., Zhang, H., Jordan, M., Gonzalez, J. E., & Stoica, I. (2024). Chatbot Arena: An Open Platform for Evaluating LLMs by Human Preference. ICML 2024, PMLR 235, 8359-8388.

[4] Dubois, Y., Galambosi, B., Liang, P., & Hashimoto, T. B. (2024). Length-Controlled AlpacaEval: A Simple Way to Debias Automatic Evaluators. COLM 2024.

[5] Gu, J., et al. (2026). A Survey on LLM-as-a-Judge. The Innovation, 7(6), 101253.

[6] IEEE Standards Association (2025). IEEE Standard for Algorithmic Bias Considerations (IEEE Std 7003-2024).

[7] Kim, S., Shin, J., Cho, Y., Jang, J., Longpre, S., Lee, H., Yun, S., Shin, S., Kim, S., Thorne, J., & Seo, M. (2024a). Prometheus: Inducing Fine-grained Evaluation Capability in Language Models. ICLR 2024.

[8] Kim, S., Suk, J., Longpre, S., Lin, B. Y., Shin, J., Welleck, S., Neubig, G., Lee, M., Lee, K., & Seo, M. (2024b). Prometheus 2: An Open Source Language Model Specialized in Evaluating Other Language Models. EMNLP 2024, 4334-4353.

[9] Li, R., Patel, T., & Du, X. (2024). PRD: Peer Rank and Discussion Improve Large Language Model based Evaluations. TMLR.

[10] Li, X., Li, C., Liu, W., Long, Y., Huang, Y., & Liu, Y. (2026). Cross-Model Adjudication for Bias Mitigation in Large Language Models. IEEE Journal of Selected Topics in Signal Processing, 20(2), 177-191.

[11] Liu, Y., Iter, D., Xu, Y., Wang, S., Xu, R., & Zhu, C. (2023). G-Eval: NLG Evaluation using GPT-4 with Better Human Alignment. EMNLP 2023, 2511-2522.

[12] Lünstedt, J., & Schlippe, T. (2025). Mitigating Bias in Large Language Models Leveraging Multi-Agent Scenarios. 2025 7th International Conference on Natural Language Processing (ICNLP), 14-18.

[13] Panickssery, A., Bowman, S. R., & Feng, S. (2024). LLM Evaluators Recognize and Favor Their Own Generations. NeurIPS 2024, 37, 68772-68802.

[14] Park, J., Jwa, S., Meiying, R., Kim, D., & Choi, S. (2024). OffsetBias: Leveraging Debiased Data for Tuning Evaluators. Findings of EMNLP 2024, 1043-1067.

[15] Sollenberger, Z., Patel, J., Munley, C., Jarmusch, A., & Chandrasekaran, S. (2024). LLM4VV: Exploring LLM-as-a-Judge for Validation and Verification Testsuites. SC24-W, 1885-1893.

[16] Tan, S., Zhuang, S., Montgomery, K., Tang, W., Cuadron, A., Wang, C., Popa, R., & Stoica, I. (2025). JudgeBench: A Benchmark for Evaluating LLM-Based Judges. ICLR 2025, 63277-63303.

[17] Wang, P., Li, L., Chen, L., Cai, Z., Zhu, D., Lin, B., Cao, Y., Kong, L., Liu, Q., Liu, T., & Sui, Z. (2024). Large Language Models are not Fair Evaluators. ACL 2024, Vol. 1, 9440-9450.

[18] Wang, Y., Yu, Z., Zeng, Z., Yang, L., Wang, C., Chen, H., Jiang, C., Xie, R., Wang, J., Xie, X., Ye, W., Zhang, S., & Zhang, Y. (2024). PandaLM: An Automatic Evaluation Benchmark for LLM Instruction Tuning Optimization. ICLR 2024.

[19] Ye, J., Wang, Y., Huang, Y., Chen, D., Zhang, Q., Moniz, N., Gao, T., Geyer, W., Huang, C., Chen, P.-Y., Chawla, N. V., & Zhang, X. (2025). Justice or Prejudice? Quantifying Biases in LLM-as-a-Judge. ICLR 2025.

[20] Zhao, R., Zhang, W., Chia, Y. K., Xu, W., Zhao, D., & Bing, L. (2025). Auto-Arena: Automating LLM Evaluations with Agent Peer Battles and Committee Discussions. ACL 2025, Vol. 1, 4440-4463.

[21] Zheng, L., Chiang, W.-L., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., Lin, Z., Li, Z., Li, D., Xing, E. P., Zhang, H., Gonzalez, J. E., & Stoica, I. (2023). Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. NeurIPS 2023, Datasets and Benchmarks Track, 36, 46595-46623.