

When Critiques Do Not Change Decisions: DSAR for Threshold-Aligned LLM Reviewers

Zijun Mao¹, Peng Zhang^{2,†}, Liu Lun¹, Haozhe Yang¹

¹The International Joint Institute of Tianjin University, Fuzhou, Tianjin University

²School of Computer Science and Technology, Tianjin University
pzhang@tju.edu.cn

Abstract

Large language models (LLMs) are increasingly deployed as automated reviewers whose numeric scores directly determine accept/reject outcomes, yet whether those scores faithfully reflect the models’ own critiques remains underexplored. We identify *threshold-alignment failure*: an LLM reviewer may clearly articulate serious weaknesses while still assigning scores above the decision boundary, rendering its critique inconsequential. We introduce DSAR (Dimension-Specialist Audit and Revision), a training-free protocol that decomposes evidence elicitation across dimension specialists, applies severity-coded adversarial audit, and enforces deterministic score corrections for critical findings before thresholding. On PropAlign-240 under a shared Qwen2.5-72B-Instruct backbone, six standard reviewer protocols keep most human-rejected proposals above the decision threshold, yielding false-positive rates of 89.2–100.0%. DSAR raises reject recall to 45.9% and achieves the lowest MAE (0.334), with all decision-level gains statistically significant (McNemar $p < 0.008$). Ablations are consistent with severity-based audit correction, rather than longer reasoning or additional agents, as the main source of the improvement. These findings argue that LLM reviewers should be evaluated not only by critique quality, but by whether critiques translate into threshold-consistent decisions.

1 Introduction

Large language models (LLMs) now routinely serve as automated reviewers and scoring agents (Liu and Shah, 2023; Liang et al., 2024; Zheng et al., 2023; Chan et al., 2024; Liu et al., 2023). In many screening settings, however, the central question is not whether a model can write a plausible review, but whether its scores support a fixed accept/reject threshold. Once a threshold determines

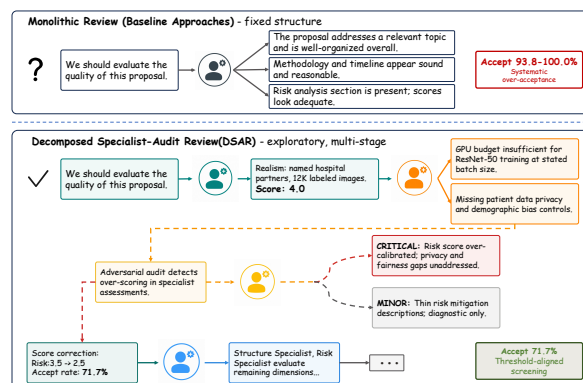


Figure 1: Threshold-alignment failure: the gap between evidence and decision. Standard reviewer protocols identify substantive weaknesses but still keep scores above the accept/reject threshold, yielding 89.2–100.0% FPR on Gold-60. DSAR closes this gap by making CRITICAL evidence score-changing before thresholding, raising reject recall to 45.9%.

the outcome, the reviewer becomes a *decision interface*: critique matters only insofar as it translates into score changes that separate proposals that should pass from those that should not.

We identify a recurring yet overlooked failure in this role: *threshold-alignment failure*. An LLM reviewer may note superficial risk analysis or missing technical safeguards, yet still assign a score above the acceptance boundary—the model has recognized relevant evidence, but the evidence has not changed the decision. The failure is therefore not a lack of critique, but a failure of critique-to-score transfer. This failure is distinct from sycophantic tendencies (Sharma et al., 2024) and presentation-level judge biases (Ye et al., 2025): it is not an agreement pattern or a surface confound, but a disconnect between evidence recognition and score movement inside the review protocol.

Standard score calibration does not resolve this failure. Post-hoc methods treat scores as globally shifted and correctable with held-out labels (Platt, 2000; Guo et al., 2017), whereas threshold-

[†]Corresponding author.

alignment failure arises inside the review protocol: a serious weakness may be mentioned but never made score-changing. Chain-of-thought, self-reflection, and multi-agent debate can produce richer rationales (Wei et al., 2022; Madaan et al., 2023; Du et al., 2024), but they do not specify *when* an identified weakness must lower a score. Additional deliberation can thus improve commentary while leaving thresholded decisions unchanged—a pattern we confirm empirically across six diverse protocols.

DSAR (Dimension-Specialist Audit and Revision) is designed to make this missing link explicit. Dimension specialists elicit evidence for complementary aspects of a proposal; an adversarial auditor classifies weaknesses as CRITICAL or MINOR; and a deterministic correction rule lowers scores only when CRITICAL findings are present. The goal is not additional agents per se, but to convert severe critique into score changes before the accept/reject threshold is applied. DSAR should be read as a decision-interface intervention rather than as a general multi-agent reasoning recipe.

We evaluate on PropAlign-240, a controlled benchmark of 240 proposals with human-annotated Gold-60 verdicts, using a fixed Qwen2.5-72B-Instruct backbone, a shared threshold ($\tau=3.0$), and an identical evaluation pipeline. Six baseline protocols yield false-positive rates of 89.2–100.0% on human-rejected proposals; DSAR raises reject recall to 45.9% and achieves the lowest MAE (0.334), with all decision-level gains statistically significant (McNemar $p<0.008$). Our contributions are:

1. We identify threshold-alignment failure—a disconnect between evidence recognition and score movement—as a distinct failure mode in LLM-based screening, separate from score miscalibration and critique-quality deficits.
2. We construct PropAlign-240, a controlled diagnostic testbed with adjudicated human verdicts, and introduce DSAR, a training-free protocol that makes severe critique score-changing under a fixed threshold without requiring labeled calibration data.
3. Ablations are consistent with severity-coded audit correction as the main source of improvement, rather than longer reasoning, extra deliberation, or agent count; they identify severity detection for weak-but-present evidence as the open frontier.

2 Related Work

LLMs as reviewers and LLM-as-judge. LLMs are increasingly used to generate reviews, surface technical issues, check rubric violations, and simulate feedback (Liu and Shah, 2023; Liang et al., 2024; Jin et al., 2024; Zhu et al., 2025; D’Arcy et al., 2024; Yu et al., 2024). In parallel, LLM-as-judge research studies reference-free evaluation and documents systematic biases such as position, verbosity, and self-preference effects (Zheng et al., 2023; Chan et al., 2024; Wang et al., 2024; Laskar et al., 2024; Li et al., 2024b; Ye et al., 2025; Lai et al., 2026); earlier supervised approaches focus on review-outcome prediction (Kang et al., 2018). Together, these lines establish that LLMs can produce plausible judgments but that their judgments are vulnerable to known biases. Proposal screening, however, imposes a stricter requirement: numeric scores must remain usable under a fixed accept/reject threshold. DSAR targets this decision-interface gap, converting critique into score changes that directly affect thresholded outcomes.

Reasoning and multi-agent review protocols. Test-time reasoning methods—including chain-of-thought (Wei et al., 2022), self-consistency (Wang et al., 2023), tree-of-thought (Yao et al., 2023), self-refinement (Madaan et al., 2023), and reflexive feedback (Shinn et al., 2023)—expand the space of intermediate rationales. Multi-agent protocols further introduce debate, role specialization, or collaborative review (Du et al., 2024; Chan et al., 2024; Li et al., 2024a; Fan et al., 2026; Jin et al., 2024; D’Arcy et al., 2024). These approaches can elicit richer critiques, but richer critique does not by itself determine when a recognized weakness should lower a score. This gap is compounded by evidence that LLMs often cannot self-correct without external feedback (Huang et al., 2024) and tend toward sycophantic agreement over accuracy (Sharma et al., 2024; Perez et al., 2023). DSAR addresses the missing link directly: weaknesses are audited with severity labels, and only CRITICAL findings trigger deterministic score correction.

Calibration and thresholded decision making. Classical calibration aligns model confidence with observed outcomes through post-hoc correction (Platt, 2000; Guo et al., 2017), while recent work examines whether LLM confidence estimates and judge score distributions remain calibrated at the task level (Desai and Durrett, 2020; Kadavath et al.,

2022; Ulmer et al., 2024; Li et al., 2025). Related factuality research likewise shows that fluent text need not be reliable (Manakul et al., 2023; Min et al., 2023). These approaches address reliability after scores or outputs have been produced, often requiring labeled calibration data or method-specific correction functions. Threshold-alignment failure arises earlier: the reviewer may already identify relevant evidence yet leave it score-neutral. DSAR therefore intervenes inside the review protocol rather than tuning thresholds after the fact, producing threshold-usable scores through severity-coded audit and deterministic correction.

3 PropAlign-240 Construction

PropAlign-240 is a controlled testbed for studying threshold alignment against adjudicated human verdicts in proposal screening. The benchmark holds proposal distribution and evaluation conditions fixed while varying only the reviewer protocol. Controlled generation enables balanced domain-quality cells, avoids private institutional data, and makes quality strata and failure patterns fully reproducible.

Benchmark structure. The benchmark contains 240 English proposals across four domains (healthcare, education, agriculture, transportation), balanced over 12 domain-quality cells (4 domains $\times$ 3 quality levels $\times$ 20 proposals per cell). Candidates were generated with GPT-4o (OpenAI, 2024) using quality-stratified templates, then filtered for format errors, length outliers, and near-duplicates. To reduce model-side generator-evaluator coupling, generation, quality control, and evaluation use separate model families (GPT-4o, Claude Sonnet, and Qwen2.5-72B-Instruct). Human verdicts, rather than construction strata, serve as evaluation labels. Generation templates and filtering criteria are reported in Appendix F.

Gold-60 annotation. Gold-60 was formed by stratified sampling (five per domain-quality cell, seed 42). Five annotators each scored every proposal on the four dimensions using a shared rubric and assigned binary verdicts. Final verdicts were determined by majority vote (Appendix F).

Gold-60 contains 23 accepts and 37 rejects. Aggregate-score agreement is substantial (Krippendorff $\alpha = 0.692$, Krippendorff 2019; pairwise Spearman $\rho = 0.844$; Table 1), indicating consistent overall ranking across annotators. Dimension-

Statistic	Value
Total proposals	240
Domains	4
Domain $\times$ quality cells	4 $\times$ 3
Proposals per cell	20
Quality levels	80 / 80 / 80
Avg. proposal length	2,642 words
Gold subset	60
Annotators	5
Gold Accept / Reject	23 / 37
Gold Accept rate	38.3%
Fleiss κ (verdict)	0.433
Krippendorff α (overall)	0.692
Pairwise Spearman ρ	0.844

Table 1: PropAlign-240 statistics and human agreement on the Gold-60 annotation subset. Proposal length is counted by whitespace splitting.

level α is lower (0.571 realism, 0.604 feasibility, 0.640 completeness, and 0.378 risk adjustment), so dimension-specific analyses should be interpreted as diagnostic. Verdict-level agreement is moderate (Fleiss $\kappa = 0.433$; Fleiss 1971), partly because one annotator adopted a stricter threshold (16.7% vs. 36.7–61.7% accept rate); pairwise overall-score correlations remain high ($\rho \geq 0.744$) despite this absolute-level variation. Majority verdicts are the only labels used for primary decision evaluation; the remaining 180 proposals support distributional analyses.

4 Method

4.1 Task Formulation

Given a proposal text x , a reviewer produces four dimension scores $y_d(x) \in [1, 5]$ for realism (R), feasibility (F), completeness (C), and risk adjustment (K). The overall score is the arithmetic mean $s(x) = \frac{1}{4} \sum_d y_d(x)$, and the screening decision under threshold τ is

$$\hat{z}_\tau(x) = \mathbb{I}[s(x) \geq \tau], \quad \tau = 3.0.$$

The operational pipeline serializes this mean to two decimal places before metric computation and thresholding; all reported score metrics and decisions use that serialized value. The task is thresholded screening: all methods produce decisions via the same threshold, and the evaluation target is agreement with adjudicated human verdicts rendered under the same scoring interface (Section 3).

4.2 DSAR Overview

DSAR is designed to close the evidence-to-score gap identified in Section 2: reviewer protocols of-

ten recognize weaknesses but leave their effect on scores implicit. The protocol separates three operations that are usually collapsed into a single review step. First, *dimension specialists* elicit evidence for complementary scoring dimensions, preventing serious local gaps from being averaged away by a holistic review. Second, an *adversarial auditor* assigns each identified weakness a CRITICAL or MINOR severity label, making the strength of evidence explicit and inspectable. Third, a *deterministic correction rule* changes scores only when CRITICAL findings are present, ensuring that severe evidence becomes score-changing before thresholding. The corrected scores are then evaluated under the same threshold as all other protocols. Figure 2 illustrates the pipeline.

4.3 Dimension-Specialist Evidence Elicitation

DSAR partitions the four scoring dimensions across three specialists. Each specialist reads the full proposal but scores only its assigned dimensions. The **Tech-Specialist** scores realism and feasibility by checking methodology depth, technical plausibility, and scope–resource alignment. The **Structure-Specialist** scores completeness through a section-coverage matrix that flags missing, thin, or boilerplate content. The **Risk-Specialist** scores risk adjustment by separating explicitly addressed risks from omitted risks and evaluating mitigation specificity.

Each specialist returns a structured output with dimension scores, anchor-level labels (HIGH/MED/LOW), and key-evidence summaries. By design, specialists produce grounded, dimension-local evidence for the auditor rather than issuing accept/reject verdicts, preserving a clean separation between evidence elicitation and decision making. Output schemas are in Appendix A.

4.4 Adversarial Audit with Severity Coding

The adversarial auditor inspects the proposal together with all specialist outputs. Rather than replacing specialist scores, it emits structured findings that identify the affected dimension, severity label (CRITICAL or MINOR), supporting evidence span, and recommended score adjustment.

A CRITICAL finding indicates that a specialist score is materially unsupported by the proposal—for example, a buzzword-only methodology or a timeline that compresses a 12-month study into 3 months. A MINOR finding records a real weakness that should remain diagnostic rather than score-

changing. Only CRITICAL findings trigger correction; MINOR findings are preserved in the audit trace. The severity labels are explicitly specified in the audit interface to keep the correction logic transparent and fully reproducible without training data; complete trigger conditions are in Appendix A.

4.5 Audit-Conditioned Score Correction

Let u_d denote the pre-audit specialist score for dimension d . Each CRITICAL finding m carries an auditor-recommended adjustment $a_m < 0$ (typically -1.0 ; occasionally -1.5). Before correction, the implementation applies a fixed schema normalization $\nu(\cdot)$ that retains canonical names and maps the auditor aliases `structure`→`completeness` and `risk`→`risk_adjustment`. The deterministic correction rule then computes

$$p_d = \sum_{m=1}^M \mathbb{I}[\nu(d_m)=d] \cdot \mathbb{I}[\sigma_m=\text{CRITICAL}] \cdot |a_m|,$$

and then applying clipping and half-step snapping,

$$\hat{y}_d = \text{snap}_{0.5}(\text{clip}_{[1,5]}(u_d - p_d)).$$

Empirically, the auditor emits 404 CRITICAL findings over 240 proposals; 96.5% recommend -1.0 (390 of 404) and 3.5% recommend -1.5 . Of these findings, 335 already use canonical names and 69 use the two schema aliases above; all are normalized before correction. The corrected overall score $\hat{s} = \frac{1}{4} \sum_d \hat{y}_d$ is compared against τ .

This separates LLM-dependent severity detection from deterministic rule-based score correction.

4.6 Implementation Details

DSAR uses four LLM calls per proposal: three parallel specialist calls and one sequential audit call. The final scores are produced by a deterministic correction rule that preserves specialist scores by default and lowers a dimension score by the audit-recommended penalty only when CRITICAL evidence is present. Interface schemas, parsing rules, and the deterministic correction logic are detailed in Appendix A.

5 Experiments

5.1 Setup, Baselines, and Metrics

All methods use the same backbone, Qwen2.5-72B-Instruct (Qwen et al., 2025), served via vLLM with

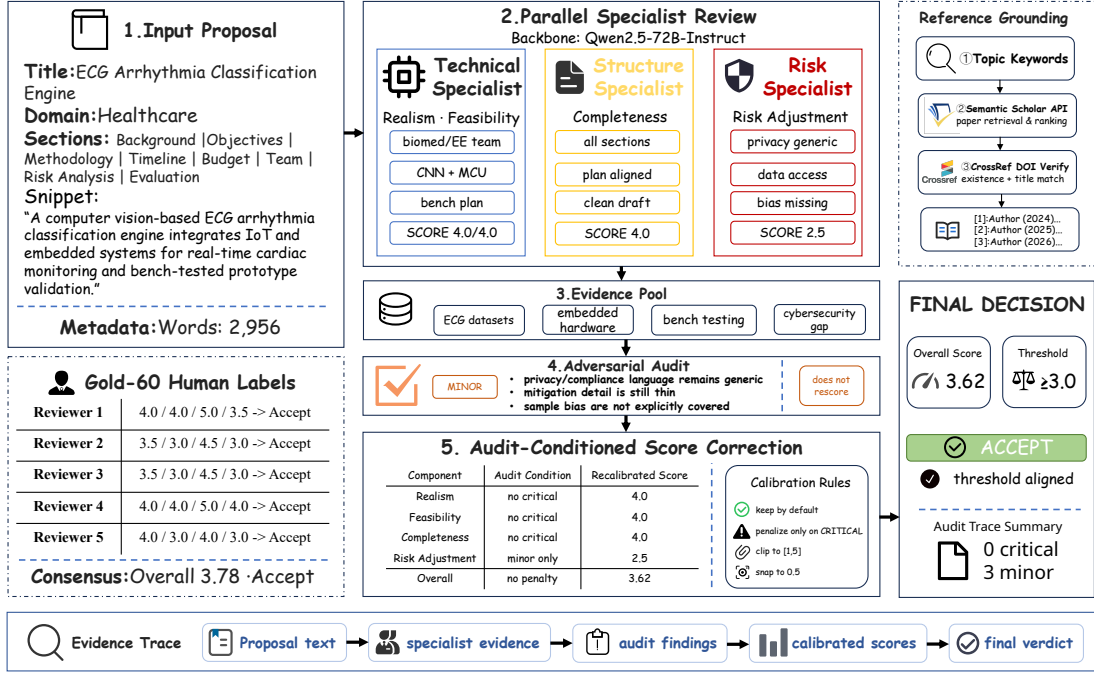


Figure 2: DSAR pipeline: specialists elicit dimension-local evidence, an adversarial auditor assigns severity labels, and a deterministic correction rule produces the final score-bearing review under a CRITICAL-only adjustment policy.

AWQ quantization, and evaluate all 240 proposals in a single stochastic replicate with an 8192-token generation cap. The shared reviewer implementations use temperature 0.1; iMAD follows its adaptive reference implementation, using temperature 0.7 for self-critique and debate agents and 0.3 for moderator/judge calls. Score-error and decision metrics are computed on Gold-60. This design prioritizes protocol comparison over backbone generality: the question is whether changing the review interface changes threshold behavior when model capacity is fixed. As a robustness check, Section 5.5 re-runs the same method family on LLaMA-3.3-70B-Instruct and Gemini-3-Flash under matched Gold-60 evaluation conditions and finds that DSAR remains competitive across backbones.

Baselines and fairness controls. We compare DSAR with six reviewer protocols: Single-Agent, Self-Reflection (Madaan et al., 2023), CoT-Agent (Wei et al., 2022), MAD (Du et al., 2024), AgentReview (Jin et al., 2024), and iMAD (Fan et al., 2026). All produce the same four dimension scores, overall score, and binary verdict under a shared rubric (1–5 scale, common anchors, and structured output schema); the prompts request 0.5-step scores. Numeric baseline outputs are range-validated but are not post-hoc rounded. DSAR’s deterministic cor-

rection explicitly snaps its score-bearing outputs to the nearest 0.5. Baselines have full access to penalize identified weaknesses, but their critique-to-score revision remains implicit. DSAR differs in making the evidence-to-score link explicit through severity-coded audit and deterministic correction.

Three controls keep the comparison aligned: all methods share the backbone, generation cap, scoring rubric, and decision rule (nominal call budgets are shown in Table 2); a shared schema places all scores on the same scale; and decisions are computed from numeric scores under the shared threshold, not from model-written verdict text. The *Generalist + Audit Correction* ablation (Table 3) isolates the effect of explicit correction by replacing specialists with a single generalist while retaining the same audit and correction rule.

Threshold and metrics. The shared threshold $\tau=3.0$ is the unified decision boundary for all methods, aligned with the rubric rule that proposals averaging below 3.0 are rejected; no method-specific tuning is used. Primary labels are adjudicated human verdicts incorporating dimension-level vetoes; decision metrics measure agreement with holistic human judgments. Score-threshold labels serve as a robustness check (Appendix Table 21). We report decision accuracy, balanced accuracy, reject recall, FPR, MAE, and accept rate. *Shift de-*

notes the difference from the Gold-60 human mean ($\mu_{GT}=3.110$), used as a score-level bias diagnostic. We emphasize reject-oriented metrics; Wilson CIs are in Appendix C.8.

5.2 Main Results

Despite spanning single-agent, self-reflection, chain-of-thought, and multi-agent debate formats, the six baselines converge to the same decision behavior: they keep most scores above the threshold and produce false-positive rates of 89.2–100.0% on human-rejected proposals (Table 2). Because DSAR uses the same backbone yet raises reject recall to 45.9%, the evidence suggests that the backbone can often identify relevant weaknesses—what differs is whether identified weaknesses are translated into score changes that cross the decision boundary. This pattern holds under both adjudicated verdicts and alternative score-threshold labels (Appendix Table 21).

DSAR raises reject recall to 45.9% (17/37), improves balanced accuracy to 70.8%, and achieves the lowest MAE (0.334); the most robust gain is in thresholded decision behavior rather than overall score error, driven by making CRITICAL audit evidence score-changing before thresholding. Error analysis reveals that the remaining false positives concentrate in the medium stratum, where weaknesses are present but superficially plausible and human annotators themselves show the lowest inter-rater agreement (Section 6.1).

5.3 Ablation: What Mechanism Produces the Gain?

Table 3 tests whether DSAR’s gains come from specialist decomposition, audit-conditioned correction, severity coding, or simply additional computation. Three deterministic ablations apply alternative aggregation or correction rules to DSAR intermediate outputs: w/o Audit uses the snapped specialist bases, w/o Severity applies a uniform -0.5 penalty to every normalized audit finding, and Score Smoothing assigns every dimension the snapped mean specialist score. Generalist + Audit evaluates the two-call DSAR variant.

The largest drop occurs when the adversarial audit is removed: reject recall falls from 45.9% to 10.8%, showing that specialist decomposition alone does not fix threshold alignment. The *Generalist* + *Audit Correction* variant—replacing specialists with a single generalist while retaining the same audit and correction rule—reaches 29.7% reject re-

call, far above baselines (0–10.8%) but below full DSAR (45.9%). This indicates that even a generalist’s output contains weakness evidence sufficient for audit-triggered correction; the bottleneck in standard protocols is the implicit critique-to-score link, not the backbone’s capacity to identify weaknesses.

Severity coding provides the next decision-relevant gain. Removing the CRITICAL/MINOR distinction reduces reject recall from 45.9% to 40.5%. Score smoothing performs worst: averaging specialist outputs without severity-coded audit nearly eliminates rejection (2.7%), showing that soft aggregation does not substitute for gated correction. The correction step itself is deterministic: the final scores are produced by the fixed CRITICAL-only rule with no LLM moderator call, so the reported pipeline uses four LLM calls (three specialists and one auditor).

Overall, the ablations support the intended mechanism: DSAR improves threshold behavior primarily through severity-coded, audit-conditioned correction, with specialist decomposition providing better evidence for the auditor. We further discuss the relation between DSAR and label-dependent calibration in Appendix E.7.

5.4 Cross-Dataset Transfer Check

To assess transfer beyond PropAlign-240, we evaluated all seven methods on Exposia (Zyska et al., 2026), a public dataset of 55 real student draft exposés. Each exposé received two instructor assessments under the same 36-criterion rubric: one during the course (Group 1) and a second after the course (Group 2) for inter-rater analysis. Because this scoring ontology differs from PropAlign’s, we first measured continuous-score agreement against the mean of the two instructor totals derived from the released criterion-level scores. The Human-human row in Table 4 reports agreement between these two assessments and serves as an inter-rater reference.

Reviewer interfaces were adapted to Exposia’s rubric while retaining the corresponding method structures and shared calibration protocol used on PropAlign-240, without tuning on Exposia labels. DSAR achieved the lowest MAE and RMSE, while its Pearson and Spearman correlations approached the Human-human reference. Relative to iMAD, the reductions in MAE and RMSE were significant ($p < 0.001$). The Pearson and Spearman differences were positive but not significant, and the ICC dif-

Method	Calls	RMSE↓	MAE↓	Shift	DecAcc↑	BalAcc↑	RejRec↑	FPR↓	Ac% ₆₀	Ac% ₂₄₀
<i>Human reference*</i>	—	—	—	—	100.0%*	100.0%*	100.0%*	0.0%*	38.3%	—
Single-Agent	1	0.604	0.497	+0.482	40.0%	51.4%	2.7%	97.3%	98.3%	96.2%
Self-Reflection	3	0.503	0.407	+0.299	40.0%	51.4%	2.7%	97.3%	98.3%	93.8%
CoT-Agent	1	0.675	0.520	+0.481	40.0%	51.4%	2.7%	97.3%	98.3%	98.3%
MAD	4	0.731	0.566	+0.509	40.0%	51.4%	2.7%	97.3%	98.3%	98.3%
AgentReview	4	0.785	0.616	+0.590	38.3%	50.0%	0.0%	100.0%	100.0%	100.0%
iMAD	5	0.616	0.503	+0.158	43.3%	53.2%	10.8%	89.2%	91.7%	95.8%
DSAR (Ours)	4	0.412	0.334	+0.010	65.0%	70.8%	45.9%	54.1%	70.0%	71.7%

Table 2: Main results on Gold-60 ($n=60$; 23 accept, 37 reject) using Qwen2.5-72B-Instruct and the shared threshold $\tau=3.0$. The italicized Human reference row reports oracle decision metrics and the human accept rate on Gold-60 only; RMSE, MAE, Shift, and Ac%₂₄₀ are undefined because humans define the evaluation reference and only Gold-60 was manually scored. Ac%₆₀ is the accept rate on Gold-60 under each method’s final decision; Ac%₂₄₀ is the accept rate over all 240 proposals and is used only as a distributional diagnostic. Bold marks best values; for the accept-rate columns, best means closest to the human accept rate (38.3%). DSAR differs significantly from all baselines on thresholded decisions (McNemar $p<0.008$; Appendix Table 14).

Variant	Calls	DecAcc↑	BalAcc↑	RejRec↑	FPR↓	Ac% ₆₀
DSAR (main)	4	65.0%	70.8%	45.9%	54.1%	70.0%
Generalist + Audit Corr.	2	56.7%	64.9%	29.7%	70.3%	81.7%
w/o Adversarial Audit	3	45.0%	55.4%	10.8%	89.2%	93.3%
w/o Severity Distinction	4	61.7%	68.1%	40.5%	59.5%	73.3%
w/ Score Smoothing	3	40.0%	51.4%	2.7%	97.3%	98.3%

Table 3: DSAR ablations on Gold-60 under the listed call budgets. Ac%₆₀ is the Gold-60 accept rate.

ference was positive and marginal under the paired permutation test ($p = 0.060$). Together, these results show that DSAR’s lower score error extends to a distinct rubric, document type, and scoring scale. We next ask whether the methods’ scores also separate lower- from higher-scoring exposés under a common split. Appendix C.7 evaluates this question using the reference median and three alternative shared cutoffs.

5.5 Cross-Backbone Validation

The main experiments fix Qwen2.5-72B-Instruct to isolate protocol effects from backbone capacity (Section 5.1). To test whether DSAR’s evidence-to-score correction is specific to that backbone, we re-run the same seven-method family on two additional backbones—LLaMA-3.3-70B-Instruct-AWQ and Gemini-3-Flash-Preview—under matched conditions: the same Gold-60 subset and five-annotator ground truth (GT overall = mean over annotators of the four dimension means), the same metrics, the same operational threshold $\tau=3.0$, and the same predicted-verdict rule ($s \geq 3.0$, with no per-quality, filename, or human-label correction).

LLaMA-3.3-70B. On LLaMA-3.3-70B (Table 5), the baselines are less uniformly over-accepting than on Qwen: Single-Agent already reaches 48.6% reject recall. Even so, DSAR at-

tains the lowest RMSE (0.499) and MAE (0.425), the highest reject recall (62.2%), the lowest FPR (37.8%), and the accept rate closest to the human reference (53.3% vs. 38.3%, against 70–90% for the baselines), while matching the best decision accuracy (68.3%, tied with Single-Agent). Its score Shift (-0.197) is among the smallest in magnitude, indicating it neither systematically over- nor under-scores. Single-Agent edges DSAR on balanced accuracy (74.3% vs. 70.2%) because it accepts every human-accepted proposal; DSAR instead rejects more genuinely weak proposals (RejRec 62.2% vs. 48.6%) at a small cost in accept-side recall—the screening-relevant trade-off emphasized in Section 6.1.

Gemini-3-Flash. On Gemini-3-Flash (Table 5), DSAR leads on every score-error and accuracy metric: the lowest RMSE (0.421) and MAE (0.338), the highest decision accuracy (80.0%) and balanced accuracy (82.1%), and the accept rate closest to the human reference (51.7% vs. 38.3%), with reject recall 73.0% and FPR 27.0%. iMAD records nominally higher reject recall (91.9%) and lower FPR (8.1%), but this reflects indiscriminate over-rejection rather than calibrated screening: it accepts only 15.0% of Gold-60 and incurs by far the worst score error (RMSE 1.064, MAE 0.920). DSAR instead combines strong reject recall with the lowest score error, separating weak from strong

Method	Pearson $\uparrow$	Spearman $\uparrow$	MAE $\downarrow$	RMSE $\downarrow$	ICC (2,1) $\uparrow$
<i>Human-human*</i>	0.481	0.449	6.764	8.695	0.358
Single-Agent	0.039	-0.066	12.691	13.895	0.078
Self-Reflection	0.154	0.165	12.655	13.849	0.080
CoT-Agent	0.434	0.291	12.327	13.441	0.096
MAD	0.155	0.150	12.673	13.864	0.080
AgentReview	0.186	0.066	11.345	12.639	0.095
iMAD	0.277	0.325	6.855	8.623	0.178
DSAR (Ours)	0.467	0.442	4.891	6.275	0.270

Table 4: Zero-shot transfer on Exposia draft exposés under the dataset’s native rubric ($n=55$). The italicized Human-human row reports inter-rater disagreement, not model error. Paired significance tests are in Appendix Table 18.

Method	Calls	RMSE $\downarrow$	MAE $\downarrow$	Shift	DecAcc $\uparrow$	BalAcc $\uparrow$	RejRec $\uparrow$	FPR $\downarrow$	Ac% ₆₀
<i>Human reference*</i>	—	—	—	—	100.0%*	100.0%*	100.0%*	0.0%*	38.3%
<i>(a) LLaMA-3.3-70B-Instruct-AWQ</i>									
Single-Agent	1	0.768	0.671	+0.671	68.3%	74.3%	48.6%	51.4%	70.0%
Self-Reflection	3	0.511	0.428	- 0.189	66.7%	73.0%	45.9%	54.1%	71.7%
CoT-Agent	1	0.830	0.742	+0.735	60.0%	67.6%	35.1%	64.9%	78.3%
MAD	4	0.540	0.468	+0.453	51.7%	60.8%	21.6%	78.4%	86.7%
AgentReview	4	0.698	0.620	+0.617	48.3%	58.1%	16.2%	83.8%	90.0%
iMAD	5	0.737	0.604	+0.381	48.3%	56.5%	21.6%	78.4%	83.3%
DSAR (Ours)	4	0.499	0.425	-0.197	68.3%	70.2%	62.2%	37.8%	53.3%
<i>(b) Gemini-3-Flash-Preview</i>									
Single-Agent	1	0.662	0.603	+0.572	53.3%	62.2%	24.3%	75.7%	85.0%
Self-Reflection	3	0.474	0.374	- 0.091	78.3%	81.6%	67.6%	32.4%	56.7%
CoT-Agent	1	0.646	0.590	+0.574	48.3%	58.1%	16.2%	83.8%	90.0%
MAD	4	0.622	0.561	+0.539	45.0%	55.4%	10.8%	89.2%	93.3%
AgentReview	4	0.699	0.655	+0.655	50.0%	59.5%	18.9%	81.1%	88.3%
iMAD	5	1.064	0.920	-0.889	66.7%	59.0%	91.9%	8.1%	15.0%
DSAR (Ours)	4	0.421	0.338	-0.156	80.0%	82.1%	73.0%	27.0%	51.7%

Table 5: Cross-backbone results on Gold-60 ($n=60$; 23 accept, 37 reject) with the (a) LLaMA-3.3-70B-Instruct-AWQ and (b) Gemini-3-Flash-Preview backbones under the shared threshold $\tau=3.0$. DSAR uses the same four-call deterministic configuration as the main experiments (§4.5). The italicized Human reference reports oracle decision metrics and the Gold-60 accept rate; its RMSE/MAE/Shift are undefined. Bold marks the best non-human value per column within each backbone panel; for Ac%₆₀, best means closest to the human accept rate (38.3%).

proposals rather than rejecting wholesale. Self-Reflection is also competitive on this backbone (DecAcc 78.3%), but DSAR remains ahead on RMSE, MAE, decision accuracy, and balanced accuracy. Under identical Gold-60, metrics, threshold, and method family, this indicates that severity-coded, audit-conditioned correction transfers across model families and is not an artifact of the Qwen backbone, complementing the cross-dataset transfer evidence on Exposia (Section 5.4).

6 Analysis

6.1 Decision Behavior by Quality Stratum

DSAR is most effective when the evidence gap is explicit. On the low construction stratum, it correctly rejects 70% of human-rejected proposals (14/20), compared with 0–15% for the baselines; on the high stratum, all methods correctly accept. The medium stratum remains the most challenging region for all automated methods: most human-

rejected medium proposals in Gold-60 are not corrected below threshold (Appendix Figure 7, left; annotator issue types in Table 24), consistent with this stratum’s borderline human agreement and the inherent difficulty of distinguishing thin-but-present content from substantive evidence.

DSAR accepts 22 of 23 human-accepted proposals (FN = 1), and this low false-negative rate should not be read as auditor leniency. Among these 22 correctly accepted proposals, 54.5% receive at least one risk-adjustment CRITICAL, 54.5% a completeness CRITICAL, and 18.2% a realism CRITICAL after dimension-name normalization. They remain above $\tau=3.0$ because the corrections are not large enough to overcome their score buffer. DSAR therefore does not reject whenever it finds flaws; it changes a decision only when severe evidence exceeds that buffer, which distinguishes it from a naive strict-scoring strategy.

This pattern clarifies the trade-off between decision improvement and dimension-level calibra-

tion. DSAR achieves the lowest overall MAE in the main comparison (Table 2); paired tests show significant improvements over five of six baselines, with Self-Reflection not reaching significance (Appendix Table 15). Because CRITICAL findings concentrate in completeness and realism, targeted corrections improve threshold behavior at the cost of some dimension-level overcorrection relative to human annotations. The overall gain arises because baselines systematically over-score weak proposals, while DSAR’s corrections move more rejected proposals below the threshold. For screening, this trade-off is favorable.

6.2 Characterizing the Weak-but-Present Evidence Frontier

DSAR does not solve threshold alignment uniformly; rather, it reveals where the remaining difficulty lies: weak-but-present evidence that satisfies surface coverage while failing substantive support.

The medium-stratum failures are mostly *weak-but-present* cases. These proposals contain the expected section or a plausible phrase, but the content is too thin to support the score. For example, a “Risk Analysis” section may list several risks with generic mitigations such as “we will monitor progress closely”, which satisfies a surface-level check while providing no actionable contingency plan.

These cases expose three distinct capabilities: mentioning a weakness, treating it as score-relevant, and making it threshold-changing. In our experiments, the auditor consistently achieves the first step but does not consistently reach the second or third, tending to classify weak-but-present evidence as MINOR rather than CRITICAL. This pattern points to severity detection—rather than penalty magnitude—as the key open challenge. DSAR performs well on missing evidence and explicit contradictions; the remaining frontier is distinguishing formally complete but substantively inadequate sections. Appendix Table 23 supports this interpretation: larger penalties reduce some medium false positives but do not address the core issue.

6.3 Why Post-hoc Threshold Tuning Is Not a Substitute

Some baselines can be partially rescued by shifting the threshold, but this requires labeled calibration data and yields method-specific boundaries. Around the predefined operational threshold

($\tau \in \{2.75, 3.00, 3.25\}$), DSAR leads all baselines at every setting (Appendix Table 17), indicating that its advantage is not an artifact of the specific $\tau=3.0$ boundary. Screening systems typically require a shared operational threshold aligned with the rubric, not per-method tuning; DSAR changes scores before thresholding, making the same threshold more usable across protocols.

This distinction also explains why MAE alone is insufficient. Self-Reflection’s overall MAE is not significantly different from DSAR’s ($p = 0.081$; Appendix Table 15), yet it rejects only 1 of 37 human-rejected proposals versus 17 for DSAR. Its errors are predominantly upward, so average score error can remain moderate while threshold behavior degrades. Decision-level metrics are therefore essential for evaluating LLM reviewers as screening interfaces.

7 Conclusion

This paper identifies and addresses threshold-alignment failure in LLM-based proposal screening: models produce substantive critiques while keeping scores above a fixed accept/reject boundary, rendering those critiques inconsequential for the decision. The underlying cause is an implicit evidence-to-score link inside the reviewer protocol, distinct from score miscalibration or critique-quality deficits.

DSAR closes this gap by converting severe audit evidence into deterministic score corrections before thresholding. On PropAlign-240, six common reviewer protocols yield false-positive rates of 89.2–100.0% on human-rejected proposals, whereas DSAR raises reject recall to 45.9% with all decision-level gains statistically significant. Ablations are consistent with severity-coded, audit-conditioned correction as the main source of this improvement rather than longer reasoning chains or additional agents.

Our analysis further identifies severity detection for weak-but-present evidence as the primary open frontier for future protocol design. More broadly, these results argue that LLM reviewer protocols should be evaluated not only by critique plausibility but by whether critiques translate into threshold-consistent decisions—a criterion that current evaluation practice largely overlooks.

Limitations

A key challenge of DSAR lies in identifying when formally present evidence is still substantively inadequate. Although the auditor often detects missing sections, explicit contradictions, and unsupported score assignments, medium-stratum proposals often contain plausible but thin descriptions that satisfy surface-level coverage while failing to provide actionable support. More effective ways to represent evidence strength—for example through stricter grounding requirements, finer-grained severity criteria, or learned severity classifiers—may further improve threshold alignment in such borderline cases. Human agreement is also uneven across dimensions (risk-adjustment $\alpha = 0.378$), so the dimension-level error analysis is diagnostic rather than a strong claim about fine-grained label reliability. In addition, each protocol is evaluated with a single stochastic replicate; paired tests quantify sampling uncertainty across proposals. PropAlign-240 is a controlled synthetic testbed whose balanced domain–quality structure makes over-acceptance observable, but real review settings may involve more heterogeneous rubrics, genre conventions, and institutional decision rules.

Ethical Considerations

All proposals in PropAlign-240 are fully synthetic and contain no real student, instructor, or institution identifiers; team and advisor fields are synthetic placeholders (Appendix F). The Gold-60 ratings were contributed by five annotators who participated voluntarily and gave informed consent for their annotations to be used in this study; the annotations were collected anonymously and record no personal information about the annotators.

DSAR is intended as a screening aid rather than a replacement for human review. Because the protocol converts audit evidence into score corrections that can change accept/reject outcomes, we do not recommend using it as the sole basis for consequential decisions about real proposals; final decisions should remain with human reviewers, with model outputs treated as inspectable evidence rather than verdicts.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (Grant Nos. 62550068 and 62276188). We also gratefully acknowledge the support of Deepexi Technology

Co., Ltd., including the Deepexi Model Research Department and Deep π Lab. This work was further supported by the project “Research on an Agentic RL Framework for Data Science and Code Generation” (2026GFW-0487).

References

- Chi-Min Chan, Weize Chen, Yusheng Su, Jianxuan Yu, Wei Xue, Shanghang Zhang, Jie Fu, and Zhiyuan Liu. 2024. [ChatEval: Towards better LLM-based evaluators through multi-agent debate](#). In *International Conference on Learning Representations*, volume 2024, pages 9079–9093.
- Mike D’Arcy, Tom Hope, Larry Birnbaum, and Doug Downey. 2024. [MARG: Multi-agent review generation for scientific papers](#). *arXiv preprint arXiv:2401.04259*.
- Shrey Desai and Greg Durrett. 2020. [Calibration of pre-trained transformers](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 295–302.
- Yilun Du, Shuang Li, Antonio Torralba, Joshua B Tenenbaum, and Igor Mordatch. 2024. [Improving factuality and reasoning in language models through multiagent debate](#). In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pages 11733–11763.
- Wei Fan, JinYi Yoon, and Bo Ji. 2026. [iMAD: Intelligent multi-agent debate for efficient and accurate LLM inference](#). In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 29403–29411.
- Joseph L. Fleiss. 1971. [Measuring nominal scale agreement among many raters](#). *Psychological Bulletin*, 76(5):378–382.
- Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. 2017. [On calibration of modern neural networks](#). In *Proceedings of the 34th International Conference on Machine Learning*, volume 70, pages 1321–1330. PMLR.
- Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Yu, Xinying Song, and Denny Zhou. 2024. [Large language models cannot self-correct reasoning yet](#). In *International Conference on Learning Representations*, volume 2024, pages 32808–32824.
- Yiqiao Jin, Qinlin Zhao, Yiyang Wang, Hao Chen, Kaijie Zhu, Yijia Xiao, and Jindong Wang. 2024. [AgentReview: Exploring peer review dynamics with LLM agents](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 1208–1226.

- Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, Scott Johnston, Sheer El-Showk, Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, and 17 others. 2022. [Language models \(mostly\) know what they know](#). *arXiv preprint arXiv:2207.05221*.
- Dongyeop Kang, Waleed Ammar, Bhavana Dalvi, Madeleine van Zuylen, Sebastian Kohlmeier, Eduard Hovy, and Roy Schwartz. 2018. [A dataset of peer reviews \(PeerRead\): Collection, insights and NLP applications](#). In *Proceedings of the 2018 conference of the North American chapter of the Association for Computational Linguistics: Human language technologies, volume 1 (long papers)*, pages 1647–1661.
- Klaus Krippendorff. 2019. *Content Analysis: An Introduction to Its Methodology*, 4th edition. SAGE Publications, Inc., Los Angeles.
- Peng Lai, Zhihao Ou, Yong Wang, Longyue Wang, Jian Yang, Yun Chen, and Guanhua Chen. 2026. [BiasScope: Towards automated detection of bias in llm-as-a-judge evaluation](#). In *International Conference on Learning Representations*, volume 2026, pages 61996–62025.
- Md Tahmid Rahman Laskar, Sawsan Alqahtani, M Saiful Bari, Mizanur Rahman, Mohammad Abdullah Matin Khan, Haidar Khan, Israt Jahan, Amran Bhuiyan, Chee Wei Tan, Md Rizwan Parvez, Enamul Hoque, Shafiq Joty, and Jimmy Xiangji Huang. 2024. [A systematic survey and critical review on evaluating large language models: Challenges, limitations, and recommendations](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 13785–13816, Miami, Florida, USA. Association for Computational Linguistics.
- Haitao Li, Junjie Chen, Qingyao Ai, Zhumin Chu, Yujia Zhou, Qian Dong, and Yiqun Liu. 2025. [CalibraEval: Calibrating prediction distribution to mitigate selection bias in LLMs-as-Judges](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16537–16552.
- Yunxuan Li, Yibing Du, Jiageng Zhang, Le Hou, Peter Grabowski, Yeqing Li, and Eugene Ie. 2024a. [Improving multi-agent debate with sparse communication topology](#). In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 7281–7294.
- Zongjie Li, Chaozheng Wang, Pingchuan Ma, Daoyuan Wu, Shuai Wang, Cuiyun Gao, and Yang Liu. 2024b. [Split and merge: Aligning position biases in LLM-based evaluators](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 11084–11108.
- Weixin Liang, Yuhui Zhang, Hancheng Cao, Binglu Wang, Daisy Yi Ding, Xinyu Yang, Kailas Vodrahalli, Siyu He, Daniel Scott Smith, Yian Yin, Daniel A. McFarland, and James Zou. 2024. [Can large language models provide useful feedback on research papers? A large-scale empirical analysis](#). *NEJM AI*, 1(8):AIoa2400196.
- Ryan Liu and Nihar B Shah. 2023. [ReviewerGPT? An exploratory study on using large language models for paper reviewing](#). *arXiv preprint arXiv:2306.00622*.
- Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. [G-Eval: NLG evaluation using GPT-4 with better human alignment](#). In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pages 2511–2522.
- Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegrefe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. [Self-refine: Iterative refinement with self-feedback](#). In *Advances in Neural Information Processing Systems*, volume 36, pages 46534–46594. Curran Associates, Inc.
- Potsawee Manakul, Adian Liusie, and Mark Gales. 2023. [SelfCheckGPT: Zero-resource black-box hallucination detection for generative large language models](#). In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pages 9004–9017.
- Sewon Min, Kalpesh Krishna, Xinxin Lyu, Mike Lewis, Wen-tau Yih, Pang Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. [FActScore: Fine-grained atomic evaluation of factual precision in long form text generation](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12076–12100.
- OpenAI. 2024. [GPT-4o system card](#).
- Ethan Perez, Sam Ringer, Kamile Lukosiute, Karina Nguyen, Edwin Chen, Scott Heiner, Craig Pettit, Catherine Olsson, Sandipan Kundu, Saurav Kadavath, Andy Jones, Anna Chen, Benjamin Mann, Brian Israel, Bryan Seethor, Cameron McKinnon, Christopher Olah, Da Yan, Daniela Amodei, and 44 others. 2023. [Discovering language model behaviors with model-written evaluations](#). In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 13387–13434, Toronto, Canada. Association for Computational Linguistics.
- John C. Platt. 2000. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In Alexander J. Smola, Peter L. Bartlett, Bernhard Schölkopf, and Dale Schuurmans, editors, *Advances in Large-Margin Classifiers*, pages 61–74. MIT Press, Cambridge, MA.
- Qwen, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan

- Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiayi Yang, Jingren Zhou, Junyang Lin, and 24 others. 2025. [Qwen2.5 technical report](#). *Preprint*, arXiv:2412.15115.
- Mrinank Sharma, Meg Tong, Tomek Korbak, David Duvenaud, Amanda Aspell, Sam Bowman, Esin DURMUS, Zac Hatfield-Dodds, Scott Johnston, Shauna Kravec, Timothy Maxwell, Sam McCandlish, Kamal Ndousse, Oliver Rausch, Nicholas Schiefer, Da Yan, Miranda Zhang, and Ethan Perez. 2024. [Towards understanding sycophancy in language models](#). In *International Conference on Learning Representations*, volume 2024, pages 110–144.
- Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. [Reflexion: Language agents with verbal reinforcement learning](#). In *Advances in Neural Information Processing Systems*, volume 36, pages 8634–8652. Curran Associates, Inc.
- Dennis Ulmer, Martin Gubri, Hwaran Lee, Sangdoo Yun, and Seong Oh. 2024. [Calibrating large language models using their generations only](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15440–15459.
- Peiyi Wang, Lei Li, Liang Chen, Zefan Cai, Dawei Zhu, Binghuai Lin, Yunbo Cao, Lingpeng Kong, Qi Liu, Tianyu Liu, and Zhifang Sui. 2024. [Large language models are not fair evaluators](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9440–9450, Bangkok, Thailand. Association for Computational Linguistics.
- Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. [Self-consistency improves chain of thought reasoning in language models](#). In *International conference on learning representations*.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022. [Chain-of-thought prompting elicits reasoning in large language models](#). In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc.
- Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. 2023. [Tree of thoughts: Deliberate problem solving with large language models](#). In *Advances in Neural Information Processing Systems*, volume 36, pages 11809–11822. Curran Associates, Inc.
- Jiayi Ye, Yanbo Wang, Yue Huang, Dongping Chen, Qihui Zhang, Nuno Moniz, Tian Gao, Werner Geyer, Chao Huang, Pin-Yu Chen, Nitesh V. Chawla, and Xiangliang Zhang. 2025. [Justice or prejudice? Quantifying biases in LLM-as-a-Judge](#). In *The Thirteenth International Conference on Learning Representations*.
- Jianxiang Yu, Zichen Ding, Jiaqi Tan, Kangyang Luo, Zhenmin Weng, Chenghua Gong, Long Zeng, Ren-Jing Cui, Chengcheng Han, Qiusi Sun, Zhiyong Wu, Yunshi Lan, and Xiang Li. 2024. [Automated peer reviewing in paper SEA: Standardization, evaluation, and analysis](#). In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 10164–10184, Miami, Florida, USA. Association for Computational Linguistics.
- Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph Gonzalez, and Ion Stoica. 2023. [Judging LLM-as-a-Judge with MT-Bench and chatbot arena](#). In *Advances in Neural Information Processing Systems*, volume 36, pages 46595–46623. Curran Associates, Inc.
- Minjun Zhu, Yixuan Weng, Linyi Yang, and Yue Zhang. 2025. [DeepReview: Improving LLM-based paper review with human-like deep thinking process](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 29330–29355.
- Dennis Zyska, Alla Rozovskaya, Ilia Kuznetsov, and Iryna Gurevych. 2026. [Exposía: Teaching and assessment of academic writing skills for research project proposals and peer feedback](#). *Preprint*, arXiv:2601.06536.

A Prompt Schemas and Output Handling

This appendix specifies the prompt interfaces used in the experiment implementation. Rather than reproducing every long instruction verbatim, we report field-complete schemas and compact role specifications that make the protocol reproducible while avoiding near-duplicate prompt prose. The appendix follows the same evidence-to-score pathway used in the main paper: all methods first map proposals into a shared reviewer interface; DSAR then adds specialist evidence elicitation, severity-coded audit, and deterministic correction. All methods were required to return parseable JSON, and the same parser and normalization rules were applied across methods.

A.1 Shared Output Format

The prompt-based baselines use a shared target schema, and DSAR normalizes its final output into the same four dimension scores and overall decision fields. This shared interface ensures that all protocols are evaluated on the same score scale and decision rule; DSAR differs only in how the scores are produced before normalization. Table 6 summarizes the shared interface and DSAR role split; detailed prompt tables follow below.

Shared Review Schema

Use. Common target JSON returned by the baseline reviewer protocols.

Normalization. DSAR uses specialist-specific intermediate schemas but maps them back to this shared final interface.

JSON SCHEMA

```
{
  "validity_screen": {
    "verdict": "accept or reject",
    "checks": {
      "structural_completeness": true,
      "domain_plausibility": true,
      "internal_consistency": true,
      "student_naturalness": true,
      "risk_awareness": true,
      "appropriate_weakness": true
    },
    "reject_reasons": []
  },
  "scores": {
    "realism": {
      "score": X,
      "justification": "..."
    },
    "feasibility": {
      "score": X,
      "justification": "..."
    },
    "completeness": {
      "score": X,
      "justification": "..."
    },
    "risk_adjustment": {
      "score": X,
      "justification": "..."
    }
  }
}
```

```
},
"overall": {
  "verdict": "accept or reject",
  "justification": "..."
},
"issue_tags": [],
"key_issues": [],
"confidence": {
  "score": X,
  "justification": "..."
}
}
```

Table 7: Shared JSON schema used by all baseline reviewers and by the final normalized DSAR output.

For readability, the schema names the initial precheck `validity_screen`; the implementation maps it to `realism_screen`, which is distinct from the realism dimension score under `scores.realism`.

A.2 DSAR Tech Specialist Prompt

DSAR Tech Specialist Prompt

Purpose. The Tech Specialist localizes evidence for realism and feasibility before any verdict is produced.

Input. Proposal text, proposal ID, and the realism checklist.

OUTPUT SCHEMA

```
{
  "specialist": "tech",
  "realism": {
    "criterion_checks": [
      "R1: ...",
      "...",
      "R5: ..."
    ],
    "anchor_match": "HIGH|MED|LOW",
    "score": X,
    "key_evidence": "..."
  },
  "feasibility": {
    "criterion_checks": [
      "F1: ...",
      "...",
      "F5: ..."
    ],
    "anchor_match": "HIGH|MED|LOW",
    "score": X,
    "key_evidence": "..."
  }
}
```

INSTRUCTION SUMMARY

Role. TECHNICAL specialist for undergraduate innovation project review.

Scope. Score **only** Realism and Feasibility; do not score other dimensions.

Realism checks. Inspect naturalness, scope plausibility, six realism checklist items, anchor match, and final score.

Feasibility checks. Inspect methodology depth, execution path, team-task match, resource availability, anchor match, and final score.

Output constraints. Return criterion-based evidence checks rather than free-form reasoning, use HIGH-/MED/LOW anchors, return JSON only, and omit mark-down fences.

Shared Reviewer Output Interface USE Common target JSON returned by the baseline reviewer protocols. DSAR normalizes its final output back into the same interface. COMPACT SCHEMA <pre>{ "validity_screen": {...}, "scores": { "realism": {"score": X, "just.": "..."}, "feasibility": {"score": X, "just.": "..."}, "completeness": {"score": X, "just.": "..."}, "risk_adjustment": {"score": X, "just.": "..."} }, "overall": {...}, "issue_tags": [], "key_issues": [], "confidence": {...} }</pre> NORMALIZATION The implementation maps validity_screen to realism_screen. <tr><td> DSAR Prompt Design Overview ROLE SPLIT Tech. Scores only Realism and Feasibility and returns criterion checks with key evidence. Structure. Scores only Completeness using a section matrix and consistency checks. Risk. Scores only Risk_Adjustment using mentioned and omitted risks. Auditor. Does not re-score; emits CRITICAL or MINOR findings with recommended adjustments. Finalizer. A deterministic correction rule produces the final scores under a fixed Critical-only adjustment policy: it starts from the specialist scores and applies the audit-recommended penalty only for CRITICAL findings. No LLM call is used in this step. CORRECTION POLICY Only CRITICAL findings change scores. Use the auditor's negative recommended_adjustment. Typical adjustment: -1.0. Severe adjustment: -1.5. Sum CRITICAL adjustments per dimension. Clip to [1.0, 5.0]. Snap to nearest 0.5. Ignore MINOR findings for score changes. OUTPUT CONSTRAINT All calls are required to return strict JSON without mark-down fences. </td></tr>	DSAR Prompt Design Overview ROLE SPLIT Tech. Scores only Realism and Feasibility and returns criterion checks with key evidence. Structure. Scores only Completeness using a section matrix and consistency checks. Risk. Scores only Risk_Adjustment using mentioned and omitted risks. Auditor. Does not re-score; emits CRITICAL or MINOR findings with recommended adjustments. Finalizer. A deterministic correction rule produces the final scores under a fixed Critical-only adjustment policy: it starts from the specialist scores and applies the audit-recommended penalty only for CRITICAL findings. No LLM call is used in this step. CORRECTION POLICY Only CRITICAL findings change scores. Use the auditor's negative recommended_adjustment. Typical adjustment: -1.0. Severe adjustment: -1.5. Sum CRITICAL adjustments per dimension. Clip to [1.0, 5.0]. Snap to nearest 0.5. Ignore MINOR findings for score changes. OUTPUT CONSTRAINT All calls are required to return strict JSON without mark-down fences.
DSAR Prompt Design Overview ROLE SPLIT Tech. Scores only Realism and Feasibility and returns criterion checks with key evidence. Structure. Scores only Completeness using a section matrix and consistency checks. Risk. Scores only Risk_Adjustment using mentioned and omitted risks. Auditor. Does not re-score; emits CRITICAL or MINOR findings with recommended adjustments. Finalizer. A deterministic correction rule produces the final scores under a fixed Critical-only adjustment policy: it starts from the specialist scores and applies the audit-recommended penalty only for CRITICAL findings. No LLM call is used in this step. CORRECTION POLICY Only CRITICAL findings change scores. Use the auditor's negative recommended_adjustment. Typical adjustment: -1.0. Severe adjustment: -1.5. Sum CRITICAL adjustments per dimension. Clip to [1.0, 5.0]. Snap to nearest 0.5. Ignore MINOR findings for score changes. OUTPUT CONSTRAINT All calls are required to return strict JSON without mark-down fences.	

Table 6: Compact overview of the shared reviewer interface and DSAR role split. The left panel shows the common JSON target used by all protocols; the right panel shows how DSAR adds specialist evidence, severity-coded audit, and deterministic correction.

Table 8: Tech specialist prompt schema. This role grounds realism and feasibility scores in criterion-level technical evidence.

A.3 DSAR Structure Specialist Prompt

DSAR Structure Specialist Prompt Purpose. The Structure Specialist localizes evidence for completeness through section coverage and internal consistency checks. Input. Proposal text and proposal ID. OUTPUT SCHEMA <pre>{ "specialist": "structure", "completeness": { "section_matrix": [{ "section": "Title Background ...", "present": true, "depth": "<depth>", "key_content": "..." }], "depth_distribution": { "detailed": 0, "adequate": 0, "thin": 0, "missing": 0 }, "consistency_issues": [...], "anchor_match": "HIGH MED LOW", "score": X, "key_evidence": "..." } }</pre>	<pre>} }</pre> INSTRUCTION SUMMARY Role. STRUCTURE specialist for undergraduate innovation project review. Scope. Score only Completeness. Required checks. Build a section-by-section matrix over Title, Background, Objectives, Methodology, Timeline, Budget, Team, Expected Outcomes, Risk Analysis, and References. Consistency checks. Count depth buckets, identify boilerplate sections, and verify budget–method–timeline consistency. Output constraints. Match the HIGH/MED/LOW anchor and return JSON only.
---	---

Table 9: Structure specialist prompt schema. This role grounds completeness scores in section coverage, depth, and consistency evidence.

A.4 DSAR Risk Specialist Prompt

DSAR Risk Specialist Prompt

Purpose. The Risk Specialist localizes evidence for risk adjustment by separating addressed risks from omitted or weakly mitigated risks.

Input. Proposal text and proposal ID.

OUTPUT SCHEMA

```
{
  "specialist": "risk",
  "risk_adjustment": {
    "risks_mentioned": [
      {
        "risk": "...",
        "mitigation": "...",
        "quality": "specific|vague|missing"
      }
    ],
    "risks_missing": ["..."],
    "mitigation_counts": {
      "specific": 0,
      "vague": 0,
      "missing": 0
    },
    "anchor_match": "HIGH|MED|LOW",
    "score": X,
    "key_evidence": "..."
  }
}
```

INSTRUCTION SUMMARY

Role. RISK specialist for undergraduate innovation project review.

Scope. Score **only** Risk_Adjustment.

Required checks. List risks explicitly mentioned in the proposal, evaluate mitigation specificity, and identify obvious omitted risks given the domain and scope.

Aggregation. Count specific, vague, and missing mitigations before assigning the anchor and score.

Output constraints. Match the HIGH/MED/LOW anchor and return JSON only.

Table 10: Risk specialist prompt schema. This role grounds risk-adjustment scores in addressed risks, omitted risks, and mitigation specificity.

A.5 DSAR Adversarial Auditor Prompt

DSAR Adversarial Auditor Prompt

Purpose. The auditor does not re-score the proposal. It checks whether specialist scores are unsupported by the proposal evidence and emits severity-coded findings.

Input. The proposal text and the three specialist outputs.

OUTPUT SCHEMA

```
{
  "audits": [
    {
      "dimension": "<dimension>",
      "specialist_score": X,
      "severity": "CRITICAL|MINOR",
      "issue": "...",
      "evidence": "...",
      "criterion_met": "... or null",
      "recommended_adjustment": X
    }
  ],
  "critical_count": 0,

```

```
    "minor_count": 0,
    "overall_calibration_check":
      "<calibration label>",
    "verdict_consistency_note": "..."
  }
}
```

INSTRUCTION SUMMARY

Role. Audit reviewer; do not re-score the proposal.

Objective. Flag over-scored judgments and identify which weaknesses should become score-changing evidence.

CRITICAL rule. Use **CRITICAL** when proposal evidence materially fails to support the specialist score, for example under buzzword-only methodology, missing required content, impossible timelines, undergraduate-infeasible scope, unsupported HIGH anchors, boilerplate sections, or missed internal contradictions.

MINOR rule. Use **MINOR** for real weaknesses that should remain diagnostic rather than score-changing.

Adjustment policy. Recommend -1.0 for one critical issue and -1.5 for multiple critical issues on the same dimension.

Output constraints. Return JSON only.

Table 11: Adversarial auditor prompt schema. The auditor emits severity-coded findings rather than replacement scores; recommended_adjustment supplies the penalty magnitude consumed by the deterministic correction rule.

A.6 DSAR Deterministic Correction Rule

DSAR Deterministic Correction Rule

Purpose. The finalizer produces the final normalized review by applying a deterministic correction rule to the specialist scores and audit report. This step uses no LLM call.

Input. Specialist scores, specialist key evidence, and the audit report.

OUTPUT FORMAT

The rule produces the shared final-review schema in Appendix A. The implementation then normalizes the final dimension scores into scores.realism, scores.feasibility, scores.completeness, and scores.risk_adjustment.

RULE SUMMARY

Default policy. Preserve specialist scores unless the audit identifies **CRITICAL** evidence.

No smoothing. Do not average across specialists or smooth scores toward a prior; score changes are audit-conditioned.

Adjustment policy. Apply only negative recommended_adjustment values from **CRITICAL** findings; ignore **MINOR** findings for score changes.

Canonical-name rule. Before correction, the fixed aliases structure→completeness and risk→risk_adjustment are normalized to the canonical scoring dimensions. Scores are clipped to [1.0, 5.0] and snapped to the nearest 0.5.

Verdict rule. Apply the rubric verdict rule for the textual verdict.

Table 12: Deterministic correction rule summary. The reported Qwen implementation uses this deterministic correction output as the final score-bearing review; the rule is fully algorithmic and adds no LLM call beyond the three specialists and one auditor.

A.7 Parser and Post-processing Rules

All reported runs use a fixed parser and post-processing sequence. The parser removes mark-down fences, attempts standard JSON parsing, extracts the first balanced JSON block when needed, and applies light cleanup for smart quotes, trailing commas, and zero-width characters. Two deterministic syntax repairs address observed content-preserving errors: lifting a misplaced key-value pair from an array tail, and converting arithmetic-style values such as "score": $3.0 + 0.5 = 3.5$ to the final value after the last equals sign. The external `json_repair` fallback was disabled in the reported Qwen2.5 runs, so no main result depends on that package.

At the field level, the four-call DSAR scorer assigns the rubric-neutral value of 3.0 when a dimension entry is not represented in canonical numeric form. These cases are processed under the same predefined scoring pipeline used for metric computation. General score normalization constrains numeric scores to the valid range; DSAR’s deterministic correction additionally clips scores to $[1.0, 5.0]$ and snaps them to 0.5 increments. The final Qwen DSAR scores are then serialized to two decimal places.

A.8 Example JSON Outputs

Figure 3 shows representative DSAR intermediate outputs: a specialist evidence record and an audit finding that triggers deterministic correction.

B Detailed DSAR Workflow

B.1 Workflow Description

DSAR implements the evidence-to-score pathway described in the main paper. First, three specialists run in parallel and localize evidence for their assigned dimensions. Second, an adversarial auditor consumes the proposal and specialist outputs, then emits CRITICAL or MINOR findings without replacing the specialist scores. Third, a deterministic correction rule reads the specialist outputs and the audit report, then produces the final dimension scores, overall score, and thresholded decision under a fixed no-smoothing, CRITICAL-only adjustment policy. This finalizer is fully algorithmic and

adds no LLM call, so the reported pipeline uses four LLM calls in total.

B.2 Deterministic Correction Rule

For reproducibility, we restate the deterministic correction rule in implementation-aligned notation. Let $u_d = \text{Score}(H, d)$ be the inherited specialist score for dimension $d \in \mathcal{D}$, where H denotes the specialist outputs. Each CRITICAL audit finding m targeting dimension d carries a recommended adjustment $a_m < 0$; in the reported Qwen setting, the auditor is instructed to choose from $\{-1.0, -1.5\}$, and 96.5% of the 404 observed CRITICAL adjustments are -1.0 (390 findings; the remaining 14 are -1.5). Positive adjustments are zeroed, and MINOR findings do not change scores. A fixed normalization ν maps structure to completeness and risk to risk_adjustment before score correction. The per-dimension penalty is

$$p_d = \sum_{m=1}^M \mathbb{I}[\nu(d_m)=d] \cdot \mathbb{I}[\sigma_m=\text{CRITICAL}] \cdot |a_m|.$$

DSAR then applies bounded correction and half-step normalization,

$$\tilde{y}_d = \text{clip}_{[1,5]}(u_d - p_d), \quad \hat{y}_d = \text{snap}_{0.5}(\tilde{y}_d).$$

The final overall score and operational decision are

$$\hat{s} = \frac{1}{|\mathcal{D}|} \sum_{d \in \mathcal{D}} \hat{y}_d, \quad \hat{z}_\tau = \mathbb{I}[\hat{s} \geq \tau], \quad \tau = 3.0.$$

In the reported traces, the auditor emits at most one CRITICAL finding per proposal per original dimension name; this is an observed property, not a hard constraint.

B.3 Pseudocode

Algorithm 1 gives the deterministic correction procedure. The reported Qwen main results use the four-call deterministic pipeline: three parallel specialist calls and one adversarial audit call, with the final scores produced by the deterministic CRITICAL-only correction rule.

B.4 Call Budget and Parallelism

DSAR uses four LLM calls in the reported Qwen implementation: three independent specialist calls and one adversarial audit call. The specialist calls can be executed in parallel, while the auditor is sequential. The final scores are produced by a deterministic correction rule that applies audit-recommended penalties only when CRITICAL evidence is present.

Specialist Output Example

```
{
  "specialist": "tech",
  "realism": {
    "criterion_checks": [
      "R1: project-specific need",
      "R5: MED+0.5"
    ],
    "anchor_match": "MED",
    "score": 3.5,
    "key_evidence":
      "named site and team size"
  },
  "feasibility": {
    "criterion_checks": [
      "F1: methods named",
      "F5: MED"
    ],
    "anchor_match": "MED",
    "score": 3.0,
    "key_evidence":
      "pipeline present; data access vague"
  }
}
```

Audit and Correction Example

```
{
  "audits": [
    {
      "dimension": "completeness",
      "specialist_score": 3.5,
      "severity": "CRITICAL",
      "issue": "missing evaluation plan",
      "evidence":
        "one-sentence validation section",
      "criterion_met":
        "single-line required section",
      "recommended_adjustment": -1.0
    }
  ],
  "critical_count": 1,
  "minor_count": 0
}
```

Figure 3: Representative DSAR intermediate JSON outputs. Left: specialist evidence for dimension-local scoring. Right: an audit finding whose CRITICAL severity triggers deterministic correction.

Algorithm 1 DSAR deterministic correction procedure. Specialists elicit dimension-local evidence; the auditor labels findings by severity; the correction rule normalizes dimension names and converts CRITICAL findings into score corrections before thresholding.

Input: proposal x ; dimension set $\mathcal{D}$; threshold τ

Output: final scores $\hat{\mathbf{y}}$, overall score $\hat{s}$, decision $\hat{z}_\tau$, audit trace A

```
1:  $h_{\text{tech}} \leftarrow \text{TECHSPECIALIST}(x)$ 
2:  $h_{\text{struct}} \leftarrow \text{STRUCTURESPECIALIST}(x)$ 
3:  $h_{\text{risk}} \leftarrow \text{RISKSPECIALIST}(x)$  ▷ parallel
4:  $H \leftarrow \{h_{\text{tech}}, h_{\text{struct}}, h_{\text{risk}}\}$ 
5:  $A \leftarrow \text{AUDITOR}(x, H)$ 
6: for all  $d \in \mathcal{D}$  do
7:    $u_d \leftarrow \text{EXTRACTSCORE}(H, d)$ 
8:    $p_d \leftarrow 0$ 
9:   for all  $a_m \in A$  where  $\nu(d_m)=d \wedge \sigma_m=\text{CRITICAL}$  do
10:     $p_d \leftarrow p_d + |a_m|$  ▷ apply CRITICAL-only penalty
11:   end for
12:    $\tilde{y}_d \leftarrow \text{clip}_{[1,5]}(u_d - p_d)$ 
13:    $\hat{y}_d \leftarrow \text{snap}_{0.5}(\tilde{y}_d)$ 
14: end for
15:  $\hat{s} \leftarrow \frac{1}{|\mathcal{D}|} \sum_{d \in \mathcal{D}} \hat{y}_d$ 
16:  $\hat{z}_\tau \leftarrow \mathbb{I}[\hat{s} \geq \tau]$ 
17: return  $(\hat{\mathbf{y}}, \hat{s}, \hat{z}_\tau, A)$ 
```

Runtime. In our local A800 deployment with Qwen2.5-72B-Instruct, AWQ quantization, and vLLM serving, the reported four-call DSAR run averages 80.0 s per proposal. Single-Agent averages 19.4 s, CoT-Agent 22.3 s, Self-Reflection 52.5 s, MAD 77.3 s, AgentReview 74.8 s, and iMAD 41.2 s. DSAR’s runtime reflects its specialist-audit design and structured parsing, and should be interpreted alongside its decision-level gains under the

shared threshold.

C Additional Decision Analysis

Appendix C provides decision-level checks that complement the main results. The analyses address three questions: why several baselines have identical binary metrics despite different continuous scores, whether DSAR’s decision gains are statistically reliable on Gold-60, and whether the conclusion depends on a specific threshold or labeling convention.

C.1 Understanding the Identical Binary Metrics

Several baselines in Table 2 have identical binary metrics at the shared threshold $\tau=3.0$: Single-Agent, Self-Reflection, CoT-Agent, and MAD each reach 40.0% decision accuracy and 2.7% reject recall. This does not mean the protocols behave identically. Their continuous score distributions differ, as reflected by their MAE, RMSE, and Shift values; however, their scores remain far enough above the threshold that each method rejects only one of the 37 human-rejected proposals.

The rejected instances also differ. Single-Agent, Self-Reflection, and CoT-Agent reject SYN-0143, whereas MAD rejects SYN-0055. Thus, the baselines are distinct protocols, but their thresholded outputs collapse into the same over-accepting decision pattern.

Table 13 illustrates this effect on representative High, Medium, and Low proposals: baseline scores

vary, but most remain above $\tau=3.0$, yielding false-positive decisions on the rejected examples. DSAR moves the Low-quality proposal decisively below the threshold when CRITICAL evidence is detected, and moves the Medium-quality proposal below the threshold as well (2.88, versus baseline scores of 3.25–3.75), the only method to match the human-annotated reject verdict on that example.

C.2 Paired Decision Significance Tests

Gold-60 is evaluated with exact McNemar tests because decisions are paired at the proposal level. Table 14 shows that DSAR’s decision accuracy is significantly higher than each baseline ($p<0.008$); all six comparisons also remain significant under Holm correction (maximum adjusted $p = 0.0072$). This indicates that the decision-level gain is not explained by paired sampling variation alone on Gold-60.

Comparison	b	c	McNemar p
DSAR – Single-Agent	16	1	2.75×10^{-4}
DSAR – Self-Reflection	16	1	2.75×10^{-4}
DSAR – CoT-Agent	16	1	2.75×10^{-4}
DSAR – MAD	16	1	2.75×10^{-4}
DSAR – AgentReview	17	1	1.45×10^{-4}
DSAR – iMAD	17	4	7.20×10^{-3}

Table 14: Exact McNemar tests for paired decision accuracy on Gold-60. b counts proposals where DSAR is correct and the baseline is wrong; c counts the reverse. All comparisons remain significant after Holm correction.

C.3 Paired Bootstrap Confidence Intervals

Table 15 reports paired bootstrap intervals and sign-flip permutation tests for overall MAE on Gold-60. DSAR significantly lowers overall MAE against five of six baselines, while the improvement over Self-Reflection is not significant.

Baseline	MAE _D	MAE _B	Δ	95% CI	p
<i>Overall score</i>					
Single-Agent	.334	.497	-.164	[-.260, -.067]	.002
Self-Reflection	.334	.407	-.073	[-.152, +.007]	.081
CoT-Agent	.334	.520	-.186	[-.302, -.071]	.003
MAD	.334	.566	-.233	[-.353, -.112]	<.001
AgentReview	.334	.616	-.282	[-.408, -.156]	<.001
iMAD	.334	.503	-.169	[-.273, -.067]	.003

Table 15: Paired tests for overall MAE on Gold-60 ($n=60$; $B=10,000$). $\Delta = \text{MAE}_{\text{DSAR}} - \text{MAE}_{\text{base}}$; negative values favor DSAR. CIs use percentile bootstrap intervals, and p -values use two-tailed sign-flip permutation tests. DSAR significantly improves overall MAE against five of six baselines (the improvement over Self-Reflection is not significant). DSAR is not intended to improve every dimension-level score; it prioritizes threshold-relevant correction, which may overcorrect individual dimensions where CRITICAL findings concentrate.

C.4 Confusion Matrix under the Shared Threshold

Table 16 reports confusion matrices for the most informative protocols under the shared threshold. The contrast highlights the main decision effect: DSAR increases true rejections from 1–4 under baselines to 17, while accepting 22 of 23 human-accepted proposals.

Method	TP	FP	TN	FN
Single-Agent	23	36	1	0
iMAD (best BL)	22	33	4	1
DSAR	22	20	17	1

Table 16: Confusion matrix on Gold-60 under $\tau=3.0$. DSAR primarily improves true rejections while preserving accepted proposals.

C.5 Threshold Sweep

Table 17 varies the operational threshold across $\tau \in \{2.75, 3.00, 3.25\}$, centered on the rubric-defined boundary. We restrict the sweep to thresholds within ± 0.25 of the operational threshold because the rubric rejects proposals whose average score falls below 3.0; thresholds far from this boundary (e.g., $\tau=3.50$) would constitute post-hoc retuning rather than a meaningful local sensitivity test. DSAR leads at all three thresholds.

C.6 Paired Exposia Comparison Against iMAD

Table 18 reports paired DSAR–iMAD deltas with bootstrap confidence intervals and paired permutation tests. The most stable result is error reduction: DSAR significantly lowers MAE and RMSE

Method	SYN-0077 (High)	SYN-0104 (Med)	SYN-0137 (Low)
<i>Human Truth</i>	<i>3.40 (Accept)</i>	<i>3.13 (Reject)</i>	<i>2.43 (Reject)</i>
Single-Agent	3.75 (Accept)	3.62 (Accept)	3.25 (Accept)
Self-Reflection	3.62 (Accept)	3.25 (Accept)	3.00 (Accept)
CoT-Agent	3.75 (Accept)	3.50 (Accept)	3.50 (Accept)
MAD	3.75 (Accept)	3.75 (Accept)	3.75 (Accept)
AgentReview	3.75 (Accept)	3.75 (Accept)	3.62 (Accept)
iMAD	3.12 (Accept)	3.25 (Accept)	3.12 (Accept)
DSAR (Ours)	3.38 (Accept)	2.88 (Reject)	2.25 (Reject)

Table 13: Continuous overall scores for representative proposals across quality strata. Baselines assign different scores but often remain above $\tau=3.0$, producing identical accept decisions on rejected proposals; DSAR moves examples below the threshold when CRITICAL evidence is detected.

τ	Single-Agent	Self-Reflect	CoT-Agent	MAD	Agent-Review	iMAD	DSAR
2.75	38.3	38.3	38.3	38.3	38.3	41.7	53.3
3.00	40.0	40.0	40.0	40.0	38.3	43.3	65.0
3.25	45.0	60.0	40.0	45.0	41.7	41.7	66.7

Table 17: Decision accuracy (%) on Gold-60 across thresholds near the operational boundary. DSAR leads at all three thresholds.

($p < 0.001$). Pearson and Spearman gains are positive but not statistically significant; ICC shows a positive bootstrap interval that excludes zero but remains marginal under the paired permutation test ($p = 0.060$).

Metric	Δ	95% CI	p
Pearson	+0.190	[−0.098, 0.478]	0.369
Spearman	+0.117	[−0.199, 0.422]	0.564
MAE	−1.964	[−2.745, −1.200]	< 0.001
RMSE	−2.348	[−3.210, −1.495]	< 0.001
ICC (2,1)	+0.092	[0.024, 0.162]	0.060

Table 18: Paired Exposia comparison between DSAR and iMAD on the full $n=55$ overlap. Δ denotes DSAR−iMAD; negative values favor DSAR for MAE/RMSE, and positive values favor DSAR for correlations. Both methods score all 55 exposés, so the paired comparison uses the same proposals as the standalone transfer table.

C.7 Decision-Level Quality Separation on Exposia

Lower score error does not necessarily imply better separation between higher- and lower-scoring exposés. We therefore evaluate the same scores under a shared binary split. The reference total for each exposé is the arithmetic mean of the two instructor-group totals. Because Exposia provides no official accept/reject boundary, we use the median of the 55 reference totals as the primary cutoff, $\tau=31.0$. For consistency with the main benchmark, totals at or above τ are labeled accepted and lower totals rejected, yielding 28 reference-accepted and 27

reference-rejected cases. The cutoff is applied unchanged to every method, without method-specific tuning. Panel (b) repeats the comparison at three alternative reference-derived cutoffs. DSAR uses the four-call deterministic totals.

At $\tau=31.0$, every score from each of the five standard baselines fell at or above the cutoff. iMAD placed one of the 27 below-cutoff reference cases below the same cutoff and kept all 28 above-cutoff cases above it. DSAR placed five of the 27 below-cutoff cases below the cutoff and kept 27 of the 28 above-cutoff cases above it. Taking accept as the positive class, this gives (TP, FP, TN, FN)=(27, 22, 5, 1). These counts yield 58.2% decision accuracy, 57.5% balanced accuracy, and 18.5% reject recall. The latter exceeds iMAD (3.7%) and all five standard baselines (0.0%), although DSAR’s FPR remains high at 81.5%. At each alternative cutoff, DSAR again had the highest observed reject recall, with values ranging from 13.6% to 28.1%. The pattern is consistent: DSAR separates more low-scoring exposés than the competing protocols, but most below-cutoff cases still receive above-cutoff model scores. This result complements the continuous-score comparison while making the remaining threshold-alignment gap visible. The p -values in Table 18 concern the paired continuous-score metrics; Table 19 reports the observed decision counts and rates.

(a) Decision metrics at the shared median cutoff $\tau=31.0$ (%)					
Method	DecAcc $\uparrow$	BalAcc $\uparrow$	RejRec $\uparrow$	FPR $\downarrow$	Ac%
Single-Agent	50.9	50.0	0.0	100.0	100.0
Self-Reflection	50.9	50.0	0.0	100.0	100.0
CoT-Agent	50.9	50.0	0.0	100.0	100.0
MAD	50.9	50.0	0.0	100.0	100.0
AgentReview	50.9	50.0	0.0	100.0	100.0
iMAD	52.7	51.9	3.7	96.3	98.2
DSAR (Ours)	58.2	57.5	18.5	81.5	89.1

(b) Reject recall across shared reference-derived cutoffs (%)						
Cutoff summary	τ	Reference A/R	Five standard baselines	iMAD	DSAR	
40th percentile	29.3	33/22		0.0	4.5	13.6
Median	31.0	28/27		0.0	3.7	18.5
Mean	30.255	30/25		0.0	4.0	20.0
60th percentile	33.0	23/32		0.0	3.1	28.1

Table 19: Decision-level quality separation on Exposia ($n=55$) under shared reference-derived cutoffs. Panel (a) applies the median of the 55 reference totals to both reference and model scores; A/R denotes the resulting reference accept/reject counts. Ac% is the model accept rate, and bold marks the value closest to the 50.9% reference accept rate. Panel (b) reports reject recall at the median and three alternative shared cutoffs. The five standard baselines are Single-Agent, Self-Reflection, CoT-Agent, MAD, and AgentReview. DSAR uses the four-call deterministic total.

C.8 Decision-Metric Confidence Intervals

Table 20 reports Wilson score 95% confidence intervals for decision metrics on Gold-60. The intervals complement the point estimates in Table 2: DSAR’s reject recall interval remains above the baseline upper bounds, reinforcing the main decision-level conclusion.

Method	DecAcc [%]	RejRec [%]	FPR [%]
Single-Agent	40.0 [28.6, 52.6]	2.7 [0.5, 13.8]	97.3 [86.2, 99.5]
Self-Reflection	40.0 [28.6, 52.6]	2.7 [0.5, 13.8]	97.3 [86.2, 99.5]
CoT-Agent	40.0 [28.6, 52.6]	2.7 [0.5, 13.8]	97.3 [86.2, 99.5]
MAD	40.0 [28.6, 52.6]	2.7 [0.5, 13.8]	97.3 [86.2, 99.5]
AgentReview	38.3 [27.1, 51.0]	0.0 [0.0, 9.4]	100.0 [90.6, 100.0]
iMAD	43.3 [31.6, 55.9]	10.8 [4.3, 24.7]	89.2 [75.3, 95.7]
DSAR	65.0 [52.4, 75.8]	45.9 [31.0, 61.6]	54.1 [38.4, 69.0]

Table 20: Wilson score 95% confidence intervals for decision metrics on Gold-60. DecAcc is computed over all 60 proposals; RejRec and FPR are computed over the 37 human-rejected proposals.

C.9 Score-Threshold vs. Adjudicated-Verdict Labels

Gold-60 verdicts include dimension-level vetoes in addition to averaged scores. To check that DSAR’s advantage is not tied to this adjudication rule, we also construct score-threshold labels by applying the same $\tau=3.0$ threshold to the human mean overall score. This relabeling yields 39 accepts and 21 rejects, compared with 23 accepts and 37 rejects under adjudicated verdicts. The 16 disagreements are all medium-quality proposals that annotators rejected despite mean scores above 3.0.

Method	Adjudicated Verdict			Score-Threshold Label		
	DAcc	RejR	FPR	DAcc	RejR	FPR
Single-Agent	40.0	2.7	97.3	66.7	4.8	95.2
Self-Reflection	40.0	2.7	97.3	66.7	4.8	95.2
CoT-Agent	40.0	2.7	97.3	66.7	4.8	95.2
MAD	40.0	2.7	97.3	66.7	4.8	95.2
AgentReview	38.3	0.0	100.0	65.0	0.0	100.0
iMAD	43.3	10.8	89.2	70.0	19.0	81.0
DSAR	65.0	45.9	54.1	81.7	66.7	33.3

Table 21: Decision metrics under adjudicated verdicts and alternative score-threshold labels. DSAR’s advantage persists under both labeling schemes and increases when labels are derived from the human mean score threshold.

Table 21 shows that DSAR’s advantage persists and becomes larger under score-threshold labels: decision accuracy rises from 65.0% to 81.7%, and reject recall rises from 45.9% to 66.7%. Baselines improve more modestly because their scores remain broadly shifted above the boundary. This alternative labeling confirms that DSAR’s gain is not an artifact of the veto-based adjudication procedure; it reflects a score distribution that better separates proposals around the operational threshold.

C.10 Audit Mechanism Statistics

Table 22 summarizes the score-bearing four-call DSAR audit traces after applying the same fixed dimension-name normalization used by the implementation. Of the 404 CRITICAL findings, 335 already carry canonical names and 69 use the aliases

structure or risk; all 404 are normalized before deterministic correction. CRIT. and MINOR count normalized findings, and N_{adj} counts proposals whose specialist base differs from the deterministic post-audit score. In these traces there is at most one normalized CRITICAL finding per proposal–dimension pair, so each CRITICAL finding produces one adjusted dimension score. Completeness receives the largest average downward shift (-0.779), followed by realism (-0.504), while feasibility and risk adjustment are revised more conservatively (-0.156 and -0.269). This pattern is consistent with DSAR’s screening behavior: targeted corrections address the broad upward bias of baseline scores, while medium-stratum cases remain challenging when weak evidence is present yet classified below the CRITICAL threshold.

Dim.	CRIT.	MINOR	N_{adj}	Pre	Post	Shift
Realism	119	53	119	3.831	3.327	-0.504
Feasibility	37	69	37	3.806	3.650	-0.156
Completeness	184	24	184	3.448	2.669	-0.779
Risk Adj.	64	163	64	3.058	2.790	-0.269
Total	404	309	404			

Table 22: Audit severity and score shifts for the four-call DSAR run over all 240 proposals. Pre and Post are the mean score-bearing specialist base and deterministic score, respectively ($n=240$ for every dimension); aliases are normalized exactly as in the implementation.

C.11 Penalty Sensitivity

Table 23 evaluates DSAR under alternative deterministic penalty schedules. The default row reports the auditor-recommended schedule and the reported Qwen DSAR metrics; other rows vary only the correction magnitudes while holding specialist scores, the dimension normalization ν , and the set of CRITICAL findings fixed.

The results support two conclusions. First, DSAR’s default penalty schedule is already MAE-optimal among the tested variants (0.334); fixed uniform penalties trade off error for decision accuracy in either direction, e.g., a fixed -1.50 penalty raises decision accuracy (68.3%) at the cost of higher MAE (0.371) and a much lower accept rate (50.4% vs. 71.7%). Second, penalty magnitude is not the main remaining bottleneck: decision accuracy stays at 65.0% across the -0.75 and -1.00 schedules and both targeted variants. Larger penalties reduce some medium false positives (MFP falls from 14 to 12 at -1.50), but they do not eliminate them, because many weak-but-present failures de-

pend on whether the auditor marks the evidence as CRITICAL in the first place.

Variant	Mean ₂₄₀	MAE↓	DAcc↑	DF1↑	Ac% ₂₄₀	MFP
Default recommendation	3.11	0.334	65.0%	0.677	71.7%	14
CRIT = -0.50	3.33	0.389	50.0%	0.605	83.8%	16
CRIT = -0.75	3.21	0.345	65.0%	0.677	73.3%	14
CRIT = -1.00	3.12	0.337	65.0%	0.677	72.5%	14
CRIT = -1.50	2.91	0.371	68.3%	0.655	50.4%	12
Feas./Risk str.	3.11	0.335	65.0%	0.677	71.7%	14
Compl. cap	3.09	0.349	65.0%	0.677	71.7%	14

Table 23: Penalty sensitivity analysis for DSAR. The default row uses auditor-recommended adjustments; fixed-penalty rows replace those magnitudes while holding specialist scores, the dimension normalization ν , and the set of CRITICAL findings constant. Mean₂₄₀ is the mean overall score across all 240 proposals; DF1 is decision F1 on Gold-60; MFP denotes accepted proposals among the 17 human-rejected medium proposals in Gold-60.

D Cross-Backbone Validation Details

For the cross-backbone supplement we run DSAR in the same four-call deterministic configuration as the main experiments (§4.5): three dimension specialists, one adversarial auditor, and the deterministic correction rule, with no LLM moderator call. This configuration is fully deterministic in its correction step and therefore isolates the audit mechanism from backbone-specific free-text behavior. Inclusion follows the main protocol: records that pass standard validation or contain complete reviews with recoverable off-grid scores are included. Cross-backbone runs cover Gold-60 only, and the accept rate is best when closest to the human reference rate (38.3%).

Takeaway. Across both backbones the baselines no longer collapse into the near-universal over-acceptance observed on Qwen (Table 2); the strongest baseline differs by backbone (Single-Agent on LLaMA, Self-Reflection on Gemini), so DSAR’s margin over the best baseline is correspondingly smaller than on Qwen. Nonetheless, DSAR is the only method that is simultaneously at or near the top of both error-based (RMSE/MAE) and decision-based (reject recall, FPR) metrics on both backbones, without the catastrophic over-acceptance of the Qwen baselines or the over-rejection of iMAD on Gemini.

E Additional Visualizations

E.1 Protocol Design Space

Figure 4 organizes reviewer protocols along three axes—how evidence is elicited, how criticism is produced, and how scores are formed—next to the decision-interface properties that thresholded screening depends on. The protocols we compare differ mainly along the first two axes while leaving the third implicit, so a sharper critique need not move a score across τ ; DSAR fixes the third axis with severity-coded audit and deterministic penalty rules, which is what turns its criticism into a decision-relevant score change.

Reviewer Protocol Components		Decision Interface Properties	
Evidence Elicitation	Single-Pass Holistic Assessment	Score-Scale Portability	Threshold Alignment
	Multi-Round Self-Revision		
	Structured Chain-of-Thought		
	Multi-Agent Debate		
	Dimension-Local Specialization		
Criticism Mechanism	Role-Based Decomposition	Evidence Inspectability	Audit Traceability
	Implicit Inline Critique		
	Self-Reflection Loop		
	Peer Argumentation		
	Selective Debate Triggering		
Score Formation	Severity-Coded Adversarial Audit	Severity Distinction	Anti-Smoothing Guarantee
	Cross-Dimension Inspection		
	Decision-Relevant Criticism		
	Direct End-to-End Scoring		
	Consensus-Based Averaging		
	Chair-Mediated Aggregation	Dimension Locality	Deterministic Score Changes
	Score Smoothing / Blending		
	Audit-Conditioned Correction	Cross-Dimension Linking	Zero-Calibration Operation
	Deterministic Penalty Rules		

Figure 4: Protocol design space. DSAR combines dimension-local evidence elicitation, severity-coded audit, and audit-conditioned correction to support threshold-aligned decision interfaces. Standard protocols often improve one axis (e.g., adding self-reflection or debate) while leaving score formation implicit; DSAR yields decision-interface properties directly relevant to thresholded screening: evidence inspectability, severity distinction, deterministic score changes, and threshold alignment.

E.2 Accept Rate and Score Distribution

Figures 5 and 6 visualize the main decision-level pattern. Baselines cluster in a near-universal-acceptance region with scores generally above $\tau=3.0$; DSAR shifts the score distribution closer to the human reference mean and reduces the accept rate, producing higher decision accuracy under the same threshold.

E.3 Medium Boundary Cases

Medium-stratum proposals account for most remaining DSAR false positives: DSAR is more reliable when evidence is missing or contradictory, but weak-but-present evidence can still appear sufficient unless the auditor marks it as CRITICAL.

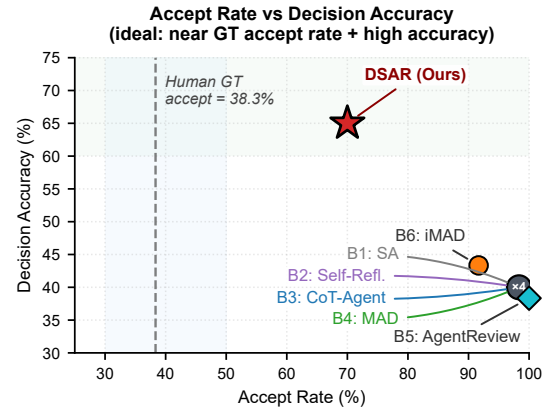


Figure 5: Accept rate vs. decision accuracy on Gold-60. Baselines cluster in a high-acceptance, low-accuracy region; DSAR moves toward lower over-acceptance and higher accuracy.

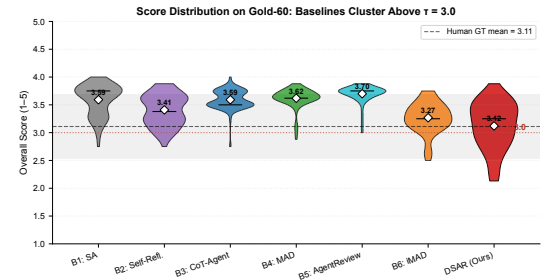


Figure 6: Overall score distributions on Gold-60. Baseline scores cluster above $\tau=3.0$; DSAR shifts closer to the human mean and creates more separation around the decision boundary.

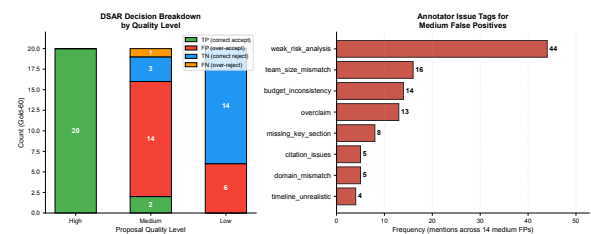


Figure 7: Medium boundary-case analysis on Gold-60. Left: DSAR decisions by quality level. Right: annotator issue tags among medium-stratum false positives, showing that remaining errors concentrate in weak-but-present evidence.

Issue type	Count	Boundary pattern
Weak risk analysis	14	Generic mitigation labeled MINOR
Team-size mismatch	7	Task list appears plausible
Budget inconsistency	4	Cross-dimension link not triggered
Overclaim	4	Ambition remains partially supported
Timeline unrealistic	1	Surface plan appears feasible

Table 24: Annotator-tagged issue types among DSAR’s 14 medium-stratum false positives on Gold-60. **Count** is the number of proposals for which at least two of the five annotators assigned the tag; the rows are the five such tags with a recurring boundary pattern. Figure 7 (right) reports mention counts for the same annotations summed over all five annotators, and its values are therefore larger.

E.4 Quality-Level Audit Gradient

Figure 8 shows how the adversarial audit responds across construction quality levels. Realism and completeness exhibit clear monotonic gradients from High to Low, indicating that the auditor detects stronger evidence gaps as proposal quality decreases. Risk adjustment follows the opposite profile, with more CRITICAL findings in the High stratum, showing that construction quality does not uniformly determine risk-audit severity and motivating the weak-but-present risk analysis in Section 6.1.

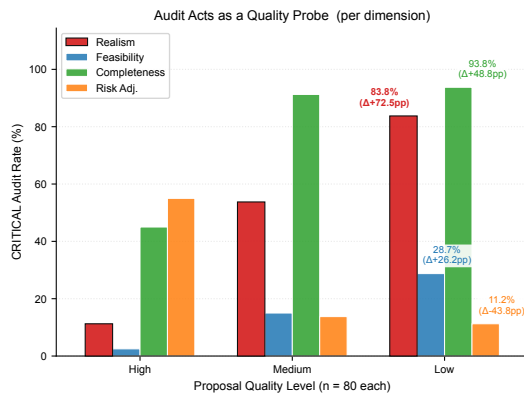


Figure 8: CRITICAL audit rate by quality level and dimension in the four-call DSAR run. Realism and completeness increase from High to Low, while risk adjustment follows the opposite profile, indicating that audit severity is dimension-specific rather than a direct proxy for construction stratum.

E.5 Diagnostic Resolution Comparison

Figure 9 compares the two conditions as dimension $\times$ quality-level score maps. In the one-call anchor-calibrated no-specialist condition, realism,

feasibility, and completeness collapse onto a single value within each quality level ($\Delta_{RFC}=0$ at all three levels), so the dimension labels carry no independent information and only the overall level separates strata; the four-call DSAR condition keeps the three dimensions apart at every level, with the widest spread in the Medium stratum, where accept/reject decisions are actually contested.

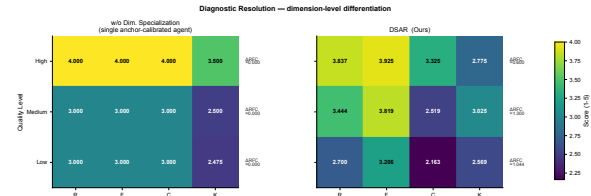


Figure 9: Diagnostic resolution comparison between a one-call anchor-calibrated no-specialist condition and the four-call DSAR condition. The one-call condition is distinct from the two-call Generalist + Audit Correction ablation in Table 3 and is included only to visualize dimensional collapse. Δ_{RFC} denotes the max-min spread across realism, feasibility, and completeness within each quality level; DSAR’s values are 0.600, 1.300, and 1.044 for High, Medium, and Low strata. This figure is a representational diagnostic, not a controlled causal ablation.

E.6 Illustrative Cases

Successful correction (SYN-0063). Single-Agent assigns 3.25 and accepts. DSAR specialists initially score realism 4.0, feasibility 4.0, completeness 3.0, and risk adjustment 3.5. The auditor emits three CRITICAL findings on realism, feasibility, and structure; after normalizing structure to completeness, the deterministic rule lowers these dimensions to 3.0, 3.0, and 2.0, respectively. The final overall score is 2.88—a reject matching the human verdict.

Boundary case (SYN-0044). DSAR specialists score realism 4.0, feasibility 4.0, completeness 3.5, and risk adjustment 2.5. The auditor marks the missing Risk Analysis section as CRITICAL under the structure alias, lowering completeness to 2.5 after normalization, but labels the omitted regulatory, bias, adoption, and cybersecurity risks only MINOR. The final overall score remains 3.25 (accept), although the human verdict is reject. This illustrates the weak-but-present evidence boundary: several risks are recognized but not classified as severe enough to produce an additional threshold-changing correction.

E.7 Post-hoc Calibration Baselines

A natural comparison is whether post-hoc calibration of Single-Agent scores can recover similar threshold behavior without changing the reviewer protocol. We consider two oracle baselines that require Gold-60 labels: (1) **Mean Shift**, which translates all Single-Agent scores by $\delta = \mu_{GT} - \mu_{SA} = -0.482$, and (2) **Threshold Tuning**, which sweeps thresholds on Gold-60 and selects the best decision-accuracy threshold ($\tau^* = 3.63$). Table 25 reports the results.

Method	$\tau/\text{corr.}$	DAcc $\uparrow$	BalAcc $\uparrow$	RejRec $\uparrow$	FPR $\downarrow$	MAE $\downarrow$
Single-Agent	$\tau=3.0$	40.0%	51.4%	2.7%	97.3%	0.497
+ Mean Shift	$\delta=-0.482$	68.3%	74.3%	48.6%	51.4%	0.309
+ Threshold Tune	$\tau^*=3.63$	81.7%	84.3%	73.0%	27.0%	0.497
DSAR	$\tau=3.0$	65.0%	70.8%	45.9%	54.1%	0.334

Table 25: Label-dependent calibration baselines on Gold-60. These rows are not operational baselines: both use Gold-60 labels that are unavailable under the fixed-threshold protocol setting. They are included to separate label-dependent score calibration from protocol-level evidence-to-score correction.

These baselines clarify the distinction between calibration and protocol design. Mean Shift reaches 68.3% decision accuracy with lower MAE (0.309), but requires Gold-60 labels to compute the correction; DSAR achieves 65.0% without any labeled data, demonstrating that protocol-level correction can approach label-dependent calibration performance. Threshold Tuning reaches 81.7% decision accuracy by selecting a method-specific threshold on the evaluation set, illustrating that given oracle access to labels, simple calibration can be highly effective. DSAR instead makes severe evidence score-changing before thresholding, without per-deployment gold labels, while preserving an inspectable audit trace for each correction. The comparison demonstrates that DSAR’s value lies in producing more threshold-usable scores without labeled calibration data, not in outperforming oracle post-hoc methods.

F Controlled Generation and Dataset Details

Appendix F documents how PropAlign-240 was constructed to support controlled protocol comparison. The goal is not to simulate every feature of real institutional review, but to create a reproducible proposal-screening testbed in which domain, quality stratum, and failure patterns are balanced and inspectable.

F.1 Use of Private Seed Documents

PropAlign-240 was informed by three non-releasable real proposal documents used only as private genre references. We abstracted section structure, proposal style, and recurring review-relevant patterns from these seeds, but did not copy factual content, project topics, institution-specific details, or identifiable text spans into the benchmark or the limited public demo examples. Retained proposals were additionally screened for private identifiers and seed leakage.

F.2 Generation Control Schema

PropAlign-240 is a synthetic English proposal-review testbed generated for controlled reviewer-protocol evaluation. The generation configuration specifies 300 candidate proposal cards across four domains, three quality levels, and 25 candidates per domain–quality cell. Candidate proposals were generated with GPT-4o; soft quality control used a separate Claude Sonnet configuration; and all reported protocol evaluation used Qwen2.5-72B-Instruct. This separation reduces generator–evaluator coupling.

Generation templates control domain, quality stratum, and expected failure patterns, but not target numeric scores or accept/reject labels. A quality-control stage retained 240 proposals, with 20 proposals per domain–quality cell and 80 per quality stratum. Hard filters enforce structural validity and data safety; soft filters score realism, feasibility, completeness, and clarity for within-stratum selection. High-quality candidates require complete structure and few major weaknesses; medium candidates are plausible but contain partial methods, thin evaluation, or weak risk analysis; low candidates remain readable but include more substantial gaps such as unrealistic scope, missing methods, budget–timeline mismatch, or overclaiming. These soft-QC scores are construction controls, not evaluation labels. Seed leakage refers to verbatim or near-verbatim reuse of private seed templates or source snippets.

Table 26 summarizes the controlled features and expected failure patterns for each quality stratum.

Quality	Controlled features	Expected failure
High	Clear need; feasible scope; complete plan; explicit mitigation	Few critical issues
Medium	Plausible idea; partial methods; thin evaluation or risk analysis	Borderline accept/reject
Low	Unrealistic scope; missing method; budget/-time mismatch; over-claiming	Frequent critical issues

Table 26: Generation control schema for synthetic quality strata. The strata guide controlled proposal construction but are not used as evaluation labels.

F.3 Quality-Control Filters

The retained proposals passed automatic filters for Markdown structure, required section coverage, extreme length, budget arithmetic, timeline coverage, team-size alignment, domain–technology compatibility, seed leakage, duplicate or near-duplicate content, and absence of private identifiers. Candidates whose realized content clearly conflicted with their assigned construction cell were removed. These filters make the quality strata interpretable as controlled experimental conditions, not hidden gold labels.

F.4 Gold-60 Sampling and Human Validation

After screening, we selected five proposals from each domain–quality cell using stratified uniform sampling with random seed 42, yielding a balanced Gold-60 subset. Five annotators scored each proposal on realism, feasibility, completeness, and risk adjustment, assigned an accept/reject verdict, and added issue tags. Human verdicts, not construction labels or soft-QC scores, serve as the adjudicated labels for evaluation.

Table 27 shows that the construction strata are well separated under human scoring: High proposals receive the highest overall scores and 100% accept rate, Medium proposals form the intended borderline region, and Low proposals are consistently rejected.

Metric	High	Med.	Low	KW test	Grad.
Overall score	3.741	3.203	2.388	$H=51.96, p<10^{-11}$	1.354
Accept rate	100%	15%	0%	—	—

Table 27: Human validation of Gold-60 construction strata. Human scores separate quality strata while remaining balanced across domains (Domain CV = 0.008; $p = 0.988$).

F.5 Auxiliary Judge Validation

Although the main comparison fixes Qwen2.5-72B-Instruct as the evaluation backbone, we also ran a separate Claude Sonnet 4.5 rubric judge as a supplementary benchmark-quality check. Gold-60 was split into 36 calibration and 24 held-out samples under the same stratified metadata, and the judge was compared against the five-annotator Gold labels. As shown in Table 28, the auxiliary judge tracks human score ranking well, while verdict agreement is weaker than rank correlation. We therefore use it only as supplementary validation of benchmark structure, not as a replacement for human adjudication or as a competing reviewer protocol.

Metric	Cal. (36)	Held (24)	All (60)
Cohen’s κ	0.571	0.396	0.502
Decision accuracy	72.2%	66.7%	70.0%
Spearman ρ (overall)	0.916	0.854	0.908
MAE (overall)	0.324	0.312	0.319
ECE	—	0.160	—
Brier score	—	0.228	—

Table 28: Auxiliary Claude Sonnet 4.5 rubric-judge validation against Gold-60. The judge tracks human score rankings well, but human adjudication remains the evaluation reference.