Feedback-to-Rubrics: Can We Learn Expert Criteria from Inline Comments?

Anonymous ACL submission

Abstract

Large language models (LLMs) are increasingly used for writing and review support, but their usefulness depends on context-dependent criteria, such as expert preferences or organization-specific conventions, that are often tacit, undocumented, and difficult to elicit directly. We propose a problem setting for learning reusable natural-language rubrics from accumulated inline comments on artifacts such as human-written or LLM-generated drafts. Our method infers rubrics from these comments and iteratively refines them by observing comment-wise mismatches between rubric-conditioned predictions and reference comments. We evaluate the proposed method in real-world review settings and in controlled settings with reference rubrics. These results show that inline comments can be distilled into reusable rubrics that support comment prediction, rubric understanding, and automatic artifact revision.

1 Introduction

Large language models (LLMs) are increasingly used to support writing and review (Lee et al., 2024), but their usefulness is limited when they lack context-dependent criteria, such as expert preferences or organization-specific conventions. A rubric makes such criteria explicit and guides LLM outputs toward expert judgments. In practice, however, the criteria that matter in a given application are often tacit, undocumented, and difficult to elicit directly from humans.

This difficulty has motivated work that learns such criteria from observed human judgments. For example, recent studies have inferred interpretable natural-language criteria from human behavior or preference data and evaluated them through held-out judgment prediction or reconstruction (Lin et al., 2024; Findeis et al., 2025). At the same time, studies on explicit rubrics have shown that they can improve LLM evaluation and feedback generation (Hashemi et al., 2024; Yuan et al., 2024; Kim et al., 2024). Together, these lines of work suggest a promising route: infer natural-language criteria from observed human behavior, and use those criteria as reusable conditioning signals for LLMs.

However, these approaches to rubric learning and rubric use are not directly applicable to review settings, where feedback is often provided as inline comments. Existing rubric learning studies assume relatively global judgment signals, such as satisfaction behavior over an entire dialogue or pairwise preferences between candidate responses (Lin et al., 2024; Findeis et al., 2025). In contrast, inline comments are strongly tied to local contexts and reveal which concern a reviewer raises for a local text span and at what level of specificity (Hanawa et al., 2021; Nagata, 2019). Therefore, learning a rubric from inline comments requires identifying and organizing the local criteria that govern comment behavior. Although prior work on feedback has studied inline comment generation and span localization, these approaches do not focus on rubric learning (Nagata, 2019; Hanawa et al., 2021; Hellman et al., 2020).

To address this gap, we propose *Feedback-to-Rubrics*, a new problem setting that formalizes rubric learning from inline comments. Figure 1 (Top) illustrates this setting: expert inline comments on text artifacts, such as human-written or LLM-generated drafts, are treated as observations of tacit criteria, and the goal is to organize those criteria into an explicit, reusable rubric. This setting is challenging because it requires learning a reusable rubric whose criteria remain grounded in local, span-level feedback while generalizing beyond individual comments.

We tackle this challenge by proposing a comment-prediction-based method that learns rubrics from inline comments and refines their crite-

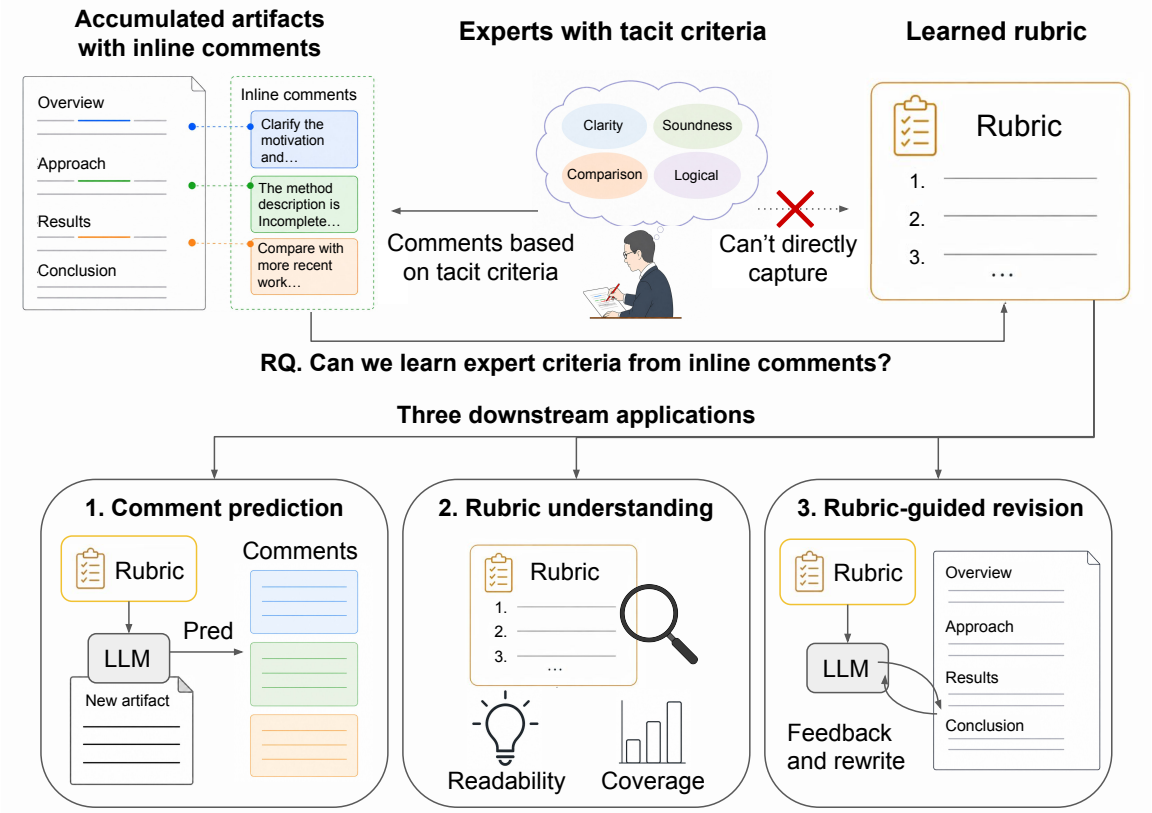

Figure 1: **Conceptual overview of Feedback-to-Rubrics and downstream applications. Top:** Feedback-to-Rubrics assumes a collection of artifacts with experts' inline comments and poses the main research question (RQ): *Can we learn expert criteria from inline comments?* The method treats these comments as observations of tacit expert criteria that are difficult to elicit directly and uses them to learn an explicit rubric. **Bottom:** The learned rubric is then used for three downstream tasks: (1) comment prediction, (2) rubric understanding, and (3) rubric-guided artifact revision.

ria using local comment mismatches. Our method infers an initial rubric from accumulated inline comments, uses it to predict comments, and iteratively refines the rubric by comparing the predicted comments with reference comments and updating the rubric based on the differences. To capture local information, we introduce a comment-wise refinement signal that links each comment-level mismatch to the criteria used to generate it, enabling updates to the granularity of individual criteria.

We evaluate the learned rubric across nine diverse tasks in the three downstream applications shown in Figure 1 (Bottom): (1) comment prediction, where LLMs conditioned on refined rubrics make more accurate comment predictions for held-out artifacts than no-rubric and retrieval-based baselines (Nagata, 2019; Lewis et al., 2020); (2) rubric understanding, where we qualitatively inspect the natural-language rubrics and quantitatively evaluate them against reference rubrics in terms of coverage and precision trade-offs; and (3) rubric-guided artifact revision, where LLMs conditioned on refined rubrics improve iterative revision over both no-rubric and initial-rubric conditions. Together, these evaluations show that rubrics learned from inline comments are useful as reusable conditioning signals across diverse tasks beyond comment prediction.

The contributions of this paper are as follows.

- We introduce Feedback-to-Rubrics, a problem setting for learning reusable natural-language rubrics from accumulated inline comments.
- We propose a comment-prediction-based method that learns rubrics from inline comments and refines them using local comment mismatches.
- We evaluate the learned rubrics in three applications: comment prediction, rubric understanding, and rubric-guided artifact revision.

2 Related Work

This work lies at the intersection of research that treats inline comments as generation targets and research that externalizes human judgment criteria as rubrics.

2.1 Inline Comment Prediction and Analysis

Prior work has studied inline comments both as generation targets and as observations of evaluation behavior. Studies on feedback comment generation model local feedback behavior, including comment prediction and span localization (Nagata, 2019; Nagata et al., 2021; Hanawa et al., 2021; Hellman et al., 2020). These studies show that feedback can be formulated around local spans or errors, rather than only as holistic artifact-level assessment. Studies on peer-review analysis instead treat comments as observations of evaluation behavior. In this domain, PeerRead introduced a large-scale peer-review dataset (Kang et al., 2018), and subsequent work has analyzed dimensions of review comments, such as substantiation, utility, and review aspect (Guo et al., 2023; Sadallah et al., 2025; Lu et al., 2025). Other work links comments to revision edits, showing that comments can also act as interventions in revision behavior (D'Arcy et al., 2024).

In contrast, we formulate Feedback-to-Rubrics as a new problem setting for learning reusable evaluation criteria from accumulated inline comments. This differs from prior work that treats comments as generation targets or objects of analysis: we use comment prediction as a testbed for learning and refining an interpretable rubric that explains expert comment behavior.

2.2 Rubric Learning for LLMs

Prior work has used rubrics as explicit intermediate representations for diagnostic evaluation and feedback, including educational assessment and automated scoring (Fiacco et al., 2023; Eltanbouly et al., 2025). These studies show that rubrics can make evaluation criteria more explicit and interpretable. Recent LLM studies further show that rubrics can be used, generated, and applied as flexible representations for controlling evaluation and feedback generation (Hashemi et al., 2024; Yuan et al., 2024; Fan et al., 2024; Gupta et al., 2025; Siro et al., 2026).

The work most closely related to ours learns explicit judgment criteria from observed human behavior or preferences. For example, one study learns interpretable satisfaction rubrics from observed dialogue behavior and validates them through held-out prediction (Lin et al., 2024). Another study compresses pairwise preference data into a constitution and evaluates it through annotation reconstruction (Findeis et al., 2025). These studies show that human judgments can be externalized as natural-language intermediate representations and used to predict or control future evaluation behavior.

However, many rubric learning studies rely on global judgment signals. In contrast, our method learns from locally grounded inline comments and uses comment-level matches as signals for refining the rubric.

3 Preliminaries: Inline Comment Prediction

This section defines the inline comment prediction setting and the evaluation protocol used by our method. Section 4 describes how we infer and update rubrics from inline comments.

Our data consist of multiple artifacts, each with multiple inline comments. Let x_i denote an artifact, q_{ij} denote the j-th target quote (a text span) in that artifact, and y_{ij} denote the corresponding reference comment. A reference comment is typically a comment written by a human reviewer, but it can also be a task-provided target comment. If multiple reference comments are associated with the same span, we treat each (q_{ij}, y_{ij}) pair as a separate instance by assigning a distinct index j. We write the full dataset as

$$\mathcal{D} = \{(x_i, \{(q_{ij}, y_{ij})\}_{j=1}^{m_i})\}_{i=1}^{N}.$$

Appendix C.4 shows examples of artifact excerpts, target quotes, and reference comments.

Following prior feedback comment generation settings (Nagata, 2019; Hanawa et al., 2021), we assume that the target quote q_{ij} is given and focus on generating a comment that matches the paired reference comment y_{ij}. We split the dataset $\mathcal{D}$ into train, validation, and test sets. The train split is used for learning, the validation split for selection, and the test split only for final evaluation.

Our evaluation measures how well generated comments align with reference comments. For each split, we compute the mean content score over an evaluation set $\mathcal{T}$ as $\frac{1}{|\mathcal{T}|}\sum_{(i,j)\in\mathcal{T}} s_{ij}$. Prior work on feedback comment generation and text generation evaluation often uses BLEU, ROUGE, or BERTScore to measure comment similarity (Nagata, 2019; Hanawa et al., 2021; Papineni et al., 2002; Lin, 2004; Zhang et al., 2020). However, inline comments make this comparison harder. The same concern can be expressed in different words,

and two comments can look similar on the surface or in embedding space while differing in concern, specificity, or actionability (Sadallah et al., 2025). Recent work suggests that generative LLMs can better capture domain-specific semantic agreement than encoder-only similarity models (Gatto et al., 2023).

Motivated by this finding, we use an LLM judge to compare the generated comment $\hat{y}_{ij}$ and the reference comment y_{ij}, conditioned on the target quote q_{ij} and the artifact x_i. The judge returns a content score s_{ij} that measures whether $\hat{y}_{ij}$ captures the same concern, granularity, and direction of explanation as y_{ij}. Because the target quote and reference comment are fixed, the judge makes a local, reference-guided semantic comparison rather than a holistic artifact-quality judgment. This also follows recent reference-guided, rubric-based evaluation and pairwise-comparison-based LLM evaluator studies showing strong alignment with human judgments in fine-grained text evaluation settings (Kim et al., 2024; Liu et al., 2024).

4 Method

This work introduces a new problem, Feedback-to-Rubrics, which aims to learn the underlying criteria used by experts from inline comments. As shown in Figure 2, our method represents the learned criteria as a rubric and improves this rubric through an iterative loop. This loop has two components: rubric-conditioned comment prediction and evaluation, which checks how well the rubric explains local comment behavior, and comment-wise rubric refinement, which updates the rubric based on those results. Appendix D lists the prompts used in our method.

4.1 Feedback-to-Rubrics

Feedback-to-Rubrics formalizes the problem of learning a reusable rubric from accumulated inline comments. Given the dataset $\mathcal{D}$ defined in Section 3, whose instances consist of artifacts x_i, target quotes q_{ij}, and reference comments y_{ij}, the goal is to infer explicit criteria that explain what concerns experts raise, when they raise them, and at what level of granularity. In this work, a rubric at round t is defined as a set of criteria $R_t = \{c_k^{(t)}\}_{k=1}^{K_t}$, where K_t denotes the number of criteria at round t. Each criterion c_k is written in natural language and contains at least four elements: (1) applicability conditions that specify when to comment in a local context, (2) the criterion text that specifies which concern to state and at what granularity, (3) boundaries that prevent confusion with nearby criteria, and (4) examples of representative target quotes and reference comments. The rubric makes explicit which concerns should be raised under which local conditions and at what granularity. As a natural-language intermediate representation of expert criteria, the rubric supports downstream uses such as rubric understanding, rubric-conditioned comment prediction, and rubric-guided artifact revision.

To solve this problem, our method first infers an initial rubric R_0 from the training split $\mathcal{D}_{\text{train}} \subset \mathcal{D}$ so that it explains the observed comment behavior. This R_0 is not assumed to be correct; it is an initial hypothesis about the latent judgment criteria expressed through the comments. It gives the method an explicit starting point that can be tested through comment prediction and revised through refinement.

4.2 Comment Prediction and Evaluation

Given a candidate rubric, comment prediction and evaluation test whether the rubric can reproduce expert comment behavior on target quotes. In this setting, a comment LLM G takes an artifact x_i, a target quote q_{ij}, and a rubric R_t as input, and generates a natural-language comment for the quote. Because the next step must know which criteria led to each generated comment, we ask the comment LLM to output both the comment text and the criteria it uses during generation. We write the prediction at round t as

$$(\hat{y}_{ij}^{(t)}, C_{ij}^{(t)}) = G(x_i, q_{ij}, R_t),$$

where $C_{ij}^{(t)} \subseteq R_t$ denotes the subset of criteria used during generation for the (i, j)-th instance.

After generation, evaluation checks whether the rubric-conditioned comment $\hat{y}_{ij}^{(t)}$ reproduces the reference comment y_{ij} and records why it succeeds or fails. An evaluation LLM J returns a content score $s_{ij}^{(t)}$ and judge reasoning $e_{ij}^{(t)}$ for each target quote:

$$(s_{ij}^{(t)}, e_{ij}^{(t)}) = J(x_i, q_{ij}, y_{ij}, \hat{y}_{ij}^{(t)}).$$

The main metric is the reproducibility of comment content. The evaluation emphasizes whether the generated comment captures the same concern, matches the reference granularity, and avoids adding unrelated critiques. Because surface overlap and embedding similarity alone cannot reliably

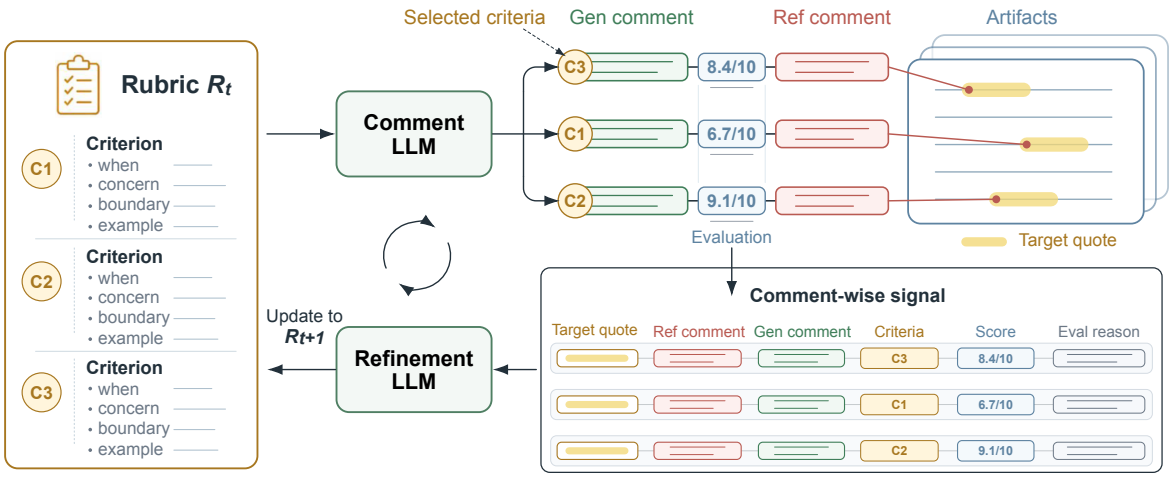

Figure 2: **Method overview.** Starting from an initial rubric R_0 inferred from inline comments, the upper part predicts comments for target quotes conditioned on the current rubric R_t and evaluates them against reference comments. The lower part then refines the rubric into R_{t+1} using comment-wise signals from these prediction/evaluation results.

support this judgment, we use a context-aware evaluation LLM and use its reasoning as a diagnostic explanation for refinement. Thus, prediction and evaluation turn each target quote into both a performance measurement and an update signal for the rubric.

4.3 Comment-Wise Iterative Rubric Refinement

Iterative rubric refinement is the update step in our proposed method: it converts prediction failures into concrete changes to the current rubric. A refinement LLM updates R_t so that the next rubric better explains the reference-comment behavior. The refinement LLM must distinguish among overly broad criteria, missing concerns, unclear applicability conditions, redundant criteria, and criteria that already align well with the reference comments. To support this decision, we provide the refinement LLM with not only the final score, but also the generated comment, the corresponding reference comment, the judge reasoning, and the criteria selected during generation.

However, if these signals are given only as separate field-wise lists, the link between a criterion and the specific comment in which it helped or failed is lost. We therefore organize them as a comment-wise refinement signal, which keeps each prediction result paired with the criteria used to generate it. For each instance at round t, we define the refinement signal as

$$h_{ij}^{(t)} = \left(q_{ij}, y_{ij}, \hat{y}_{ij}^{(t)}, s_{ij}^{(t)}, e_{ij}^{(t)}, C_{ij}^{(t)}\right), \mathcal{H}_t = \{h_{ij}^{(t)}\}.$$

Given the current rubric R_t and the comment-wise refinement signals $\mathcal{H}_t$, the refinement LLM generates the next rubric R_{t+1}. By presenting the prediction result together with the selected criteria for each comment instance, the refinement LLM can update the rubric while distinguishing local successes and failures for each criterion. Specifically, it can add a new criterion when a concern in the reference comment is missing, split or rewrite an overly broad criterion into more specific conditions, clarify the applicability conditions, boundaries, and examples of incorrectly applied criteria, and merge or remove redundant criteria. It preserves criteria that already align well with the reference comments. The updated rubric is then used in the next prediction/evaluation round, so each iteration tests the current rubric and revises it based on observed failures.

5 Experiments

We evaluate Feedback-to-Rubrics along two axes: whether our method can learn reusable rubrics from accumulated inline comments, and whether the learned rubrics are useful for downstream applications. We test these axes through three uses introduced in Figure 1: comment prediction, rubric understanding, and rubric-guided artifact revision.

- **Q1:** Does the learned rubric improve comment prediction (Section 5.2)?
- **Q2:** Can the learned rubric serve as an interpretable intermediate representation, and does refinement improve the rubric itself (Section 5.3)?
- **Q3:** Does the learned rubric guide artifact revi-

Table 1: **Ablation and baseline comparison on comment prediction.** We report mean content scores $s_{ij} \in [0, 10]$ (higher is better) averaged over five runs, with standard deviations. Our full method, which uses the comment-wise refinement signal, outperforms all baselines on average.

Method	Research proposal review	Essay review	Medical chat annotation	Bio	Chemical	Cyber	Edu	Health	Material	Avg.
Retrieval-based methods										
Top-1 retrieval (Nagata, 2019)	0.71 ±0.07	3.01 ±0.07	0.20 ±0.09	1.68 ±0.07	1.56 ±0.11	3.34 ±0.10	2.05 ±0.09	1.39 ±0.07	0.28 ±0.11	1.58
Top-3 RAG with LLM (Lewis et al., 2020)	3.05 ±0.46	8.39 ±0.09	7.08 ±0.28	2.30 ±0.28	3.12 ±0.18	4.76 ±0.22	2.26 ±0.26	2.75 ±0.18	2.06 ±0.24	3.97
Rubric-based methods (Component ablation)										
No rubric	3.00 ±0.57	7.98 ±0.12	6.62 ±0.19	0.56 ±0.14	1.76 ±0.20	1.18 ±0.23	1.37 ±0.13	1.82 ±0.35	1.79 ±0.35	2.90
+ Initial rubric	3.21 ±0.62	8.54 ±0.30	6.72 ±0.41	2.66 ±0.40	3.42 ±0.36	7.13 ±0.30	3.96 ±0.36	3.84 ±0.45	**2.43** ±0.36	4.66
+ Iterative rubric refinement	3.32 ±0.73	8.60 ±0.03	7.15 ±0.17	2.61 ±0.61	3.64 ±0.38	7.00 ±0.23	**4.11** ±0.41	4.21 ±0.48	2.35 ±0.45	4.78
+ Comment-wise refinement signal	**3.42** ±0.30	**8.71** ±0.07	**7.31** ±0.21	**2.75** ±0.48	**3.73** ±0.38	**7.43** ±0.20	3.95 ±0.34	**4.75** ±0.23	2.31 ±0.40	**4.93**

sion for new artifacts (Section 5.4)?

5.1 Datasets and Experimental Setup

Datasets. We evaluate on nine tasks, including two real-world inline-comment tasks and rubric-grounded benchmark settings. The real-world tasks test whether the method works on naturally accumulated feedback: research proposal review (Zyska et al., 2026) and essay review (Nagata et al., 2021). Publicly available inline-comment datasets suitable for research use are limited, and datasets that also include reference rubrics are particularly scarce. We therefore construct additional benchmark settings from HealthBench (Arora et al., 2025) and ExpertLongBench (Ruan et al., 2026), covering medical chat annotation and six ExpertLongBench domains: Bio, Chemical, Cyber, Edu, Health, and Material. For these benchmark settings, we use an LLM to synthesize artifacts from the original task prompts and then create rubric-grounded target-quote/comment pairs. Since ExpertLongBench provides domain-level global reference rubrics, we use it for rubric understanding and rubric-guided revision in addition to comment prediction. Across all datasets, we split at the artifact level with a 0.6/0.2/0.2 ratio for train, validation, and test. Appendix A provides construction details and split statistics.

Experimental setup. The main-text results use Gemini 3.1 Pro preview. We also run comment prediction with DeepSeek v4 Pro and GPT-5.4 low (see Appendix C.3) on a three-domain subset. To keep scoring comparable, all generated comments are judged by the same Gemini 3.1 Pro preview evaluator. Predicted comments are scored with the 0–10 content score s_{ij}, and final test results report means and standard deviations over five runs. Rubrics are learned from training comments, refined for 10 rounds using training-split prediction signals, selected by validation scores, and evaluated on the test split. We call the validation-selected rubric from the full comment-wise refinement condition the Best-val rubric. We compare rubric-conditioned generation against no-rubric generation. For retrieval baselines, we use Top-1 retrieval, following prior feedback comment generation work (Nagata, 2019), and Top-3 RAG with LLM, following retrieval-augmented generation and in-context example selection (Lewis et al., 2020; Ram et al., 2023; Rubin et al., 2022; Liu et al., 2022). Appendix B provides refinement and baseline details.

5.2 Q1: Comment Prediction

Q1 asks whether a learned rubric helps an LLM predict comments for new artifacts.

Component ablation. Table 1 reports component-ablation rows that compare four ablation settings: no rubric, an initial learned rubric, iterative refinement using aggregate field-wise signals, and comment-wise refinement that keeps each prediction paired with the criteria used for that instance. The average score increases from 2.90 without a rubric to 4.66 with an initial learned rubric, and further improves to 4.93 when comment-wise signals are used. This average trend, together with improvements in most individual task columns, indicates that updating the rubric based on alignments between generated and reference comments helps capture task-specific comment behavior. As shown in Figure 3, scores improve steadily across refinement rounds without clear overfitting. The remaining ExpertLongBench curves are reported in Appendix C.1. Appendix C.3 further shows the same overall pattern with DeepSeek v4 Pro and GPT-5.4 low backends.

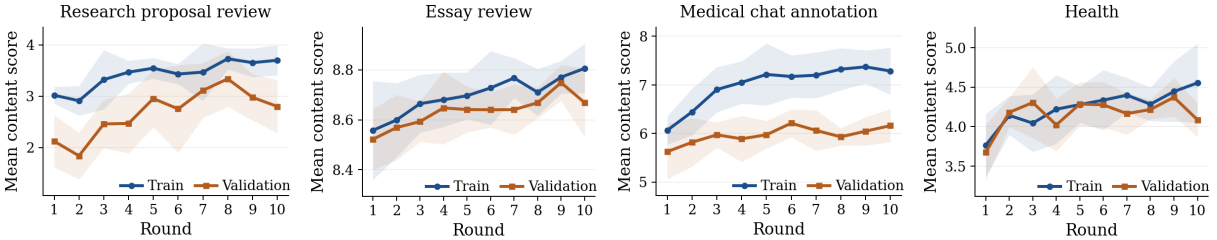

Figure 3: **Train and validation performance across refinement rounds**. Each plot shows the mean content score on the train and validation splits. In ExpertLongBench, we show the results for Health here and defer the remaining domains to Appendix C.1. In all tasks, both curves increase steadily over rounds, suggesting that refinement improves comment prediction without clear overfitting.

Table 2: **Improvement from the initial rubric to the Best-val rubric on the six ExpertLongBench domains.** Each value shows Best-val minus Initial. Scores use a 0–10 scale. H-mean denotes the harmonic mean of Recall and Precision. Overall, the Best-val rubric improves recall on average, with a slight average drop in precision and a net gain in H-mean.

Task	Recall	Precision	H-mean
Bio	**+0.40**	+0.00	**+0.22**
Chemical	**+0.84**	**+1.04**	**+0.99**
Cyber	**+0.08**	−1.06	−0.60
Edu	+0.00	−0.72	−0.46
Health	**+3.80**	−0.16	**+2.69**
Material	**+1.86**	−0.20	**+0.87**
Avg.	**+1.17**	−0.19	**+0.62**

Comparison with retrieval baselines. Table 1 also reports rows for retrieval-based baselines. Top-1 retrieval performs poorly on average (1.58), while Top-3 RAG improves to 3.97 but still falls below the initial rubric (4.66) and the full refinement setting (4.93). This suggests that learning reusable criteria from feedback is more effective for comment generation than reusing nearby past comments.

5.3 Q2: Rubric Understanding

Q2 asks whether the learned rubric functions as an inspectable natural-language representation and whether refinement improves the quality of the rubric itself.

Qualitative assessment. Figure 4 presents an example from the research proposal review task, comparing summarized initial and Best-val rubric items. We use this task for qualitative inspection because it provides a familiar setting whose criteria are easy for the research community to interpret. Full rubric texts and annotated examples are provided in Appendix C.5.

The Best-val rubric covers broader and more fine-grained criteria than the initial rubric. The initial rubric contains five items that already address core aspects of proposal quality, including missing citations, unscientific tone, overly broad research questions, unexplained theoretical frameworks, and underspecified methodology. In contrast, the Best-val rubric expands to nine items, showing more specific selection conditions. For example, refinement broadens existing criteria: citation support is extended from systems and studies to frameworks, and methodological detail is expanded to include feasibility and alignment with research questions. It also introduces new criteria for structural organization, clarity and wording, literature synthesis, and positive feedback. These observations indicate that the refinement process not only expands the coverage of criteria but also yields a rubric that captures finer-grained patterns of expert feedback that are absent from the initial rubric.

Quantitative assessment. We quantitatively evaluate the agreement between the learned rubrics and the reference rubrics on the six ExpertLongBench domains. We show the learned rubric and the reference rubric to an LLM judge and ask it to score recall and precision on a 0–10 scale. Recall measures how well the learned rubric covers important criteria in the reference global rubric. Precision measures how well the learned rubric avoids excessive or inappropriate criteria. Appendix E shows the prompt used for rubric judging.

Table 2 shows the results. Compared with the initial rubric, the Best-val rubric does not reduce recall in any domain and increases it in five of the six domains, while precision drops slightly on average. This suggests that iterative rubric refinement primarily improves coverage of reference criteria, with a precision trade-off in some domains; overall agreement still improves in four of the six domains. Therefore, the improvement in comment prediction

Example learned rubric items for research proposal review

Initial rubric (5 items)

[−] Factual claims, statistics, or references to existing systems or studies that lack citations or concrete examples.
[−] Informal, dramatic, storytelling, or marketing language in place of neutral scientific tone.
[−] Research questions that are overly broad, imprecise, or lack a specific and measurable scope.
[−] Theoretical frameworks that are named but not explained or connected to the proposed research.
[−] Methodological steps that lack concrete details such as metrics, sample sizes, or model choices.

Best-val rubric (9 items)

[−] Factual claims or references to existing systems, studies, or frameworks without citations or concrete evidence.
[−] Informal, dramatic, storytelling, conversational, or marketing language in place of neutral scientific tone.
[−] Research questions that are overly broad, imprecise, lack measurable scope, or are disjointed without a unifying focus.
[−] Theoretical frameworks or foundational concepts named but not explained or connected to the research, method, or task.
[−] Methods or study designs lacking detail, feasibility justification, or alignment with research questions.
[−] Content misplaced within sections, prematurely introduced research questions, or placeholder titles.
[−] Wording that is unclear, ambiguous, redundant, or grammatically incorrect.
[−] Literature−review sections that list papers without synthesis, comparison, or critique.
[+] Well−formulated limitations, helpful footnotes, proper research−question handling, or well−described target groups.

Figure 4: **Example learned rubric items for research proposal review.** The leading "[–]" marks negative criteria, whereas "[+]" marks positive criteria. Each line is a condensed summary of a rubric item; full texts are shown in Appendix C.5.

Table 3: **Artifact revision results on the six Expert-LongBench tasks.** The table reports changes in the number of reference rubric items satisfied after three rounds of rubric-guided artifact revision. On average, providing an explicit rubric leads to larger gains than the no-rubric condition, and using the Best-val rubric yields the highest overall improvement.

Task	No rubric	Initial	Best-val
Bio	+3.87 ±0.11	+4.88 ±0.18	**+4.95** ±0.06
Chemical	+2.92 ±0.15	+4.56 ±0.30	**+4.73** ±0.36
Cyber	+1.98 ±0.34	**+2.10** ±0.26	+2.08 ±0.20
Edu	**+0.84** ±0.16	+0.82 ±0.24	+0.78 ±0.13
Health	+6.29 ±1.59	+6.86 ±1.22	**+8.07** ±1.27
Material	+4.53 ±0.13	+4.97 ±0.25	**+5.17** ±0.20
Avg.	+3.41	+4.03	**+4.30**

is not merely due to surface-level tuning. Instead, it is consistent with a shift toward broader coverage of human judgment criteria, with a remaining precision trade-off.

5.4 Q3: Rubric-Guided Artifact Revision

Q3 asks whether a learned rubric can guide automatic artifact revision. We compare three conditions: No rubric, Initial rubric, and Best-val rubric. In the No rubric condition, the model revises the artifact without any explicit rubric. In the rubric-based conditions (Initial and Best-val), the corresponding rubric is provided to the model, and the model is instructed to revise the artifact so that it satisfies the rubric criteria. The initial artifacts are generated by an LLM to be low quality. We evaluate the revised artifacts using the reference rubric provided for each ExpertLongBench domain. Specifically, we assign one point for each reference rubric item that the revised artifact satisfies, and report the improvement in this score. We set the total number of revision rounds to three.

The results are shown in Table 3. On average, both rubric-based conditions (Initial and Best-val) improve artifact revision over the No rubric condition. In addition, when comparing the initial and Best-val rubrics, the Best-val rubric achieves a higher average improvement score. These results suggest that the rubrics learned by our framework are useful for improving artifacts themselves in a way that is aligned with task-specific preferences.

6 Conclusion

This paper introduced Feedback-to-Rubrics, a problem setting for learning reusable rubrics from naturally accumulated inline feedback. We proposed a comment-prediction-based method that infers an initial rubric from inline comments and then refines it using comment-wise mismatches between generated and reference comments. We evaluated the learned rubrics in three downstream uses: comment prediction, rubric understanding, and rubric-guided artifact revision. The learned rubrics improved comment prediction over no-rubric and retrieval-based baselines, covered reference criteria more broadly, and helped LLMs revise artifacts more effectively. These results show that inline feedback can be externalized into reusable, context-dependent criteria for evaluation, interpretation, and revision.

7 Limitations

This work has three main limitations. First, our evaluation still relies on LLM-based evaluation of predicted comments. Although recent studies support reference-guided rubric-based evaluation and pairwise LLM evaluators (Kim et al., 2024; Liu et al., 2024), agreement with human judgments in our specific setting should be further validated. Second, following established feedback comment generation settings (Nagata, 2019; Hanawa et al., 2021), we assume that target quotes are given. This choice is also motivated by the fact that local one-to-one comparison between reference and generated comments supports more reliable LLM-based judgment, as discussed above, and provides a denser update signal for rubric refinement. Extending Feedback-to-Rubrics to joint localization and commenting is future work. Third, our rubric-understanding and artifact-revision evaluations rely on constructed benchmark settings. Inline comments naturally accumulate in education, research supervision, and workplace training, for example on student writing, research proposals, and internal documents. We expect Feedback-to-Rubrics to be useful for such privately accumulated feedback. However, such data are rarely public and even more rarely paired with reference rubrics. Therefore, in this work, we partially use rubric-grounded synthetic comments for proof-of-concept evaluation. Constructing research-ready datasets with naturally occurring comments and reference rubrics remains future work.

References

Rahul K Arora, Jason Wei, Rebecca Soskin Hicks, Preston Bowman, Joaquin Quiñonero-Candela, Foivos Tsimpourlas, Michael Sharman, Meghan Shah, Andrea Vallone, Alex Beutel, Johannes Heidecke, and Karan Singhal. 2025. Healthbench: Evaluating large language models towards improved human health. *arXiv preprint arXiv:2505.08775*.

Mike D'Arcy, Alexis Ross, Erin Bransom, Bailey Kuehl, Jonathan Bragg, Tom Hope, and Doug Downey. 2024. Aries: A corpus of scientific paper edits made in response to peer reviews. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6985–7001.

Sohaila Eltanbouly, Salam Albatarni, and Tamer Elsayed. 2025. Trates: Trait-specific rubric-assisted cross-prompt essay scoring. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 20528–20543.

Zhiyuan Fan, Weinong Wang, Xing Wu, and Debing Zhang. 2024. Sedareval: Automated evaluation using self-adaptive rubrics. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 16916–16930.

James Fiacco, David Adamson, and Carolyn Rose. 2023. Towards extracting and understanding the implicit rubrics of transformer based automatic essay scoring models. In *Proceedings of the 18th workshop on innovative use of NLP for building educational applications (BEA 2023)*, pages 232–241.

Arduin Findeis, Timo Kaufmann, Eyke Hüllermeier, Samuel Albanie, and Robert Mullins. 2025. Inverse constitutional ai: Compressing preferences into principles. In *International Conference on Learning Representations*.

Joseph Gatto, Omar Sharif, Parker Seegmiller, Philip Bohlman, and Sarah Preum. 2023. Text encoders lack knowledge: Leveraging generative llms for domain-specific semantic textual similarity. In *Proceedings of the Third Workshop on Natural Language Generation, Evaluation, and Metrics (GEM)*, pages 277–288.

Yanzhu Guo, Guokan Shang, Virgile Rennard, Michalis Vazirgiannis, and Chloé Clavel. 2023. Automatic analysis of substantiation in scientific peer reviews. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10198–10216.

Taneesh Gupta, Shivam Shandilya, Xuchao Zhang, Rahul Madhavan, Supriyo Ghosh, Chetan Bansal, Huaxiu Yao, and Saravan Rajmohan. 2025. Carmo: Dynamic criteria generation for context aware reward modelling. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 2202–2261.

Kazuaki Hanawa, Ryo Nagata, and Kentaro Inui. 2021. Exploring methods for generating feedback comments for writing learning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9719–9730.

Helia Hashemi, Jason Eisner, Corby Rosset, Benjamin Van Durme, and Chris Kedzie. 2024. Llm-rubric: A multidimensional, calibrated approach to automated evaluation of natural language texts. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13806–13834.

Scott Hellman, William R Murray, Adam Wiemerslage, Mark Rosenstein, Peter Foltz, Lee Becker, and Marcia Derr. 2020. Multiple instance learning for content feedback localization without annotation. In *Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 30–40.

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.

Seungone Kim, Jamin Shin, Yejin Cho, Joel Jang, Shayne Longpre, Hwaran Lee, Sangdoo Yun, Seongjin Shin, Sungdong Kim, James Thorne, and Minjoon Seo. 2024. Prometheus: Inducing fine-grained evaluation capability in language models. In *The Twelfth International Conference on Learning Representations*.

Mina Lee, Katy Ilonka Gero, John Joon Young Chung, Simon Buckingham Shum, Vipul Raheja, Hua Shen, Subhashini Venugopalan, Thiemo Wambsganss, David Zhou, Emad A Alghamdi, and 1 others. 2024. A design space for intelligent and interactive writing assistants. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, pages 1–35.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Ying-Chun Lin, Jennifer Neville, Jack Stokes, Longqi Yang, Tara Safavi, Mengting Wan, Scott Counts, Siddharth Suri, Reid Andersen, Xiaofeng Xu, Deepak Gupta, Sujay Kumar Jauhar, Xia Song, Georg Buscher, Saurabh Tiwary, Brent Hecht, and Jaime Teevan. 2024. Interpretable user satisfaction estimation for conversational systems with large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11100–11115, Bangkok, Thailand. Association for Computational Linguistics.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, William B Dolan, Lawrence Carin, and Weizhu Chen. 2022. What makes good in-context examples for gpt-3? In *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd workshop on knowledge extraction and integration for deep learning architectures*, pages 100–114.

Yinhong Liu, Han Zhou, Zhijiang Guo, Ehsan Shareghi, Ivan Vulić, Anna Korhonen, and Nigel Collier. 2024. Aligning with human judgement: The role of pairwise preference in large language model evaluators. *Conference on Language Modeling*.

Sheng Lu, Ilia Kuznetsov, and Iryna Gurevych. 2025. Identifying aspects in peer reviews. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 6145–6167.

Ryo Nagata. 2019. Toward a task of feedback comment generation for writing learning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3206–3215.

Ryo Nagata, Masato Hagiwara, Kazuaki Hanawa, Masato Mita, Artem Chernodub, and Olena Nahorna. 2021. Shared task on feedback comment generation for language learners. In *Proceedings of the 14th International Conference on Natural Language Generation*, pages 320–324.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Ori Ram, Yoav Levine, Itay Dalmedigos, Dor Muhlgay, Amnon Shashua, Kevin Leyton-Brown, and Yoav Shoham. 2023. In-context retrieval-augmented language models. *Transactions of the Association for Computational Linguistics*, 11:1316–1331.

Jie Ruan, Inderjeet Jayakumar Nair, Shuyang Cao, Amy Liu, Sheza Munir, Micah Pollens-Dempsey, Yune-Ting Tiffany Chiang, Lucy R. Kates, Nicholas David, Sihan Chen, Ruxin Yang, Yuqian Yang, Jihyun Jasmine Gump, Tessa Bialek, Vivek S Sankaran, Margo Schlanger, and Lu Wang. 2026. Expertlongbench: Benchmarking language models on expert-level long-form generation tasks with structured checklists. In *The Fourteenth International Conference on Learning Representations*.

Ohad Rubin, Jonathan Herzig, and Jonathan Berant. 2022. Learning to retrieve prompts for in-context learning. In *Proceedings of the 2022 conference of the North American chapter of the association for computational linguistics: human language technologies*, pages 2655–2671.

Abdelrahman Sadallah, Tim Baumgärtner, Iryna Gurevych, and Ted Briscoe. 2025. The good, the bad and the constructive: Automatically measuring peer review's utility for authors. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 28991–29021.

Clemencia Siro, Pourya Aliannejadi, and Mohammad Aliannejadi. 2026. Learning to judge: Llms designing and applying evaluation rubrics. In *Findings of the Association for Computational Linguistics: EACL 2026*, pages 6371–6389.

Weizhe Yuan, Pengfei Liu, and Matthias Gallé. 2024. Llmcrit: Teaching large language models to use criteria. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 7929–7960.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Dennis Zyska, Alla Rozovskaya, Ilia Kuznetsov, and Iryna Gurevych. 2026. Exposía: Academic writing assessment of exposés and peer feedback. *arXiv preprint arXiv:2601.06536*.

Appendix

A Datasets

Table 4 reports statistics computed from the processed split files used in the experiments. The split unit is always the artifact, so multiple comments attached to the same artifact never cross train, validation, or test boundaries.

A.1 Dataset Construction Details

Across all datasets, we split at the artifact level with a 0.6/0.2/0.2 target split ratio for train, validation, and test. The table reports the resulting split counts after this artifact-level split.

Real-world inline-comment datasets. The processed research and essay data provide artifact-level inputs, target quotes, reference comments, and optional character offsets, but they do not expose per-comment annotator IDs in a uniform way. For research proposal review, the selected Exposía subset is restricted to a single reviewer. These two datasets are used as naturally occurring inline-comment settings and do not provide reference rubrics for rubric-level evaluation.

Rubric-grounded benchmark settings. HealthBench and ExpertLongBench provide rubric information that allows us to construct benchmark settings with target quotes and reference comments. HealthBench provides prompt-specific rubrics for evaluating medical responses, whereas ExpertLongBench provides domain-level global rubrics for expert-level long-form generation tasks. For this reason, rubric-level agreement and rubric-guided revision are evaluated on ExpertLongBench, whose global reference rubrics match the single shared rubric setting used by our method. For medical chat annotation, each synthetic reference comment is linked to the HealthBench criterion from which it was derived, including the criterion polarity. For each prompt, we first sample a subset of HealthBench criteria and construct synthetic artifacts designed to satisfy half of the sampled criteria while violating the remaining half. We then generate reference comments only for the violated criteria. For these constructed benchmark settings, we report the available processed fields and aggregate statistics used in the experiments. All input data had already been filtered before the experiments. We did not apply additional empty-comment filtering or quote deduplication during the reported runs. For ExpertLongBench, we construct feedback-style data from the publicly released benchmark. We use the six domains reported in Table 4: Bio, Chemical, Cyber, Edu, Health, and Material. ExpertLongBench provides expert-level long-form generation tasks and reference global rubrics, but not quote-level feedback comments. We therefore generate task outputs using the original task prompts, task-specific system prompts, and supplementary materials, and then generate quote-level reference comments grounded in the corresponding reference rubrics. Both task-output generation and reference-comment generation use `google/gemini-3-flash-preview` with medium reasoning effort and temperature 1.0. We increase output-length limits relative to the original configuration to reduce truncation. In both benchmark sources, the constructed learning instances consist of artifacts, target quotes, and reference comments. The source rubrics are used to construct benchmark instances and, for ExpertLongBench, to evaluate rubric-level agreement and rubric-guided revision; they are not used as learned attributes in the Feedback-to-Rubrics input.

Each processed input row is treated as one artifact and contains an artifact identifier, the original prompt when available, the artifact text to be reviewed, and the set of reference comments attached to the artifact. Each reference comment contains a target quote, comment text, and optional character offsets. Message-list prompts are normalized into role-and-content strings before being passed to the LLMs. When source comments include existing rubric IDs, we do not use them as learned attributes. Criterion IDs are used only as runtime traces of which learned criterion the generator cited.

Generation prompts include the full artifact, all rubric criteria, the target quote, and available start and end offsets. The generation LLM returns one structured comment and cited criterion IDs for each target quote. The implementation aligns returned IDs with the reference-comment order, stores valid criterion IDs as traces, and inserts a fallback comment if the generator omits a requested target quote.

Dataset Licenses. We used four datasets under their respective license terms. HealthBench is released under the MIT License. ExpertLongBench is licensed under CC BY-NC 4.0. Exposía is licensed under CC BY-NC 4.0 unless otherwise noted. The Essay dataset does not specify a formal license, but its distribution terms permit use primarily for non-commercial research purposes. We used all datasets solely for research purposes

Table 4: **Dataset sources, split sizes, and aggregate statistics.** Comment counts are shown as train/validation/test, with corresponding artifact counts in parentheses. Lengths are average character counts over all splits.

Data source	Dataset	Comments (Artifacts)	Comm./ art.	Quote len.	Comment len.
Exposia (Zyska et al., 2026)	Research proposal review	118/31/43 (7/2/3)	16.00	115.1	98.1
Feedback Dataset (Nagata et al., 2021)	Essay review	329/110/113 (300/100/100)	1.10	6.1	163.0
HealthBench (Arora et al., 2025)	Medical chat annotation	180/69/92 (30/10/10)	6.82	153.7	323.3
ExpertLongBench (Ruan et al., 2026)	Bio	214/74/75 (60/20/20)	3.63	177.8	210.4
	Chemical	177/58/58 (60/20/20)	2.93	176.4	188.7
	Cyber	209/62/71 (60/20/20)	3.42	108.3	192.6
	Edu	235/86/82 (60/20/20)	4.03	91.8	163.6
	Health	239/81/69 (60/20/20)	3.89	94.7	153.6
	Material	64/18/17 (30/10/10)	1.98	199.5	283.7

and complied with the conditions stated on their respective dataset pages.

B Experimental Setup Details

B.1 Main Protocol

Comment prediction follows the fixed-target-quote setting in Section 3: the model receives an artifact, a target quote, and, when applicable, a rubric, and generates one comment for that quote. Generated comments are evaluated with the content score $s_{ij} \in [0, 10]$ using the prompt in Appendix D.3. For each dataset, iterative rubric refinement runs for 10 rounds. Training-split reference comments are used for refinement, the validation split is used for round selection, and the test split is used only for final evaluation. During refinement, the model receives the current refinement signal and histories from up to three previous rounds. We run each experimental condition five times on the same split and report the mean and standard deviation over these runs.

B.2 Retrieval-Based Methods

We compare our method with the following two retrieval-based baselines.

Top-1 retrieval For each test position, we encode the target quote as a query and retrieve the most similar comment from the training set. The retrieved comment text is then copied verbatim as the prediction. This baseline is purely retrieval-based: it does not call an LLM and does not use any learned rubric. This baseline follows the retrieval setting used in prior feedback comment generation work (Nagata, 2019).

Top-3 RAG with LLM For each test position, we retrieve the top $k = 3$ most similar comments from the training set and provide them to an LLM as in-context examples. This baseline follows retrieval-augmented generation and in-context example selection, where retrieved documents or similar examples are placed in the context of a frozen or API LLM (Lewis et al., 2020; Ram et al., 2023; Rubin et al., 2022; Liu et al., 2022). The retrieved examples are included in the user prompt using the following format.

In-context Example Format

```
### Position [position index]
  target_quote: "[target quote]"
    1. retrieved comment: "[retrieved
         comment 1]", target_quote:
         "[retrieved target quote 1]",
         similarity=[score 1]
    2. retrieved comment: "[retrieved
         comment 2]", target_quote:
         "[retrieved target quote 2]",
         similarity=[score 2]
    3. retrieved comment: "[retrieved
         comment 3]", target_quote:
         "[retrieved target quote 3]",
         similarity=[score 3]
```

The Top-3 RAG with LLM baseline uses the same fixed-position prompting setup as the main generation method described in Appendix D, with two modifications. First, the system instruction states that the prompt includes comments retrieved by similarity from the reference data. Second,

the user prompt contains an additional `Retrieved Comments from Reference Data` section that shows the retrieved comments. The LLM is instructed to use the retrieved comments as behavioral references for concern scope, tone, and granularity. It is also explicitly instructed not to copy a retrieved comment verbatim unless the retrieved comment and the test position concern the same underlying issue.

B.3 LLM Calls

The main nine-task experiments use Google's `gemini-3.1-pro-preview` for rubric generation, rubric refinement, comment generation, and evaluation. For the main Gemini calls, we set the temperature to 1.0 and the thinking level to `low`. The LLM judge uses `gemini-3.1-pro-preview` for all reported evaluations, including the additional model-backend experiments. The additional model-backend experiments use DeepSeek v4 Pro and GPT-5.4 low for comment prediction on a three-domain subset, while keeping the same evaluation judge and scoring prompt.

Embeddings are computed with `gemini-embedding-2` (https://ai.google.dev/gemini-api/docs/models/gemini-embedding-2). We use 3072-dimensional embeddings, which is the default output size of the model. The resulting vectors are cast to `float32` and L2-normalized before retrieval.

The target quote, meaning the span of the source text for which a comment should be generated, is used as the retrieval query. Each training-set feedback comment is used as a retrieval document. Following the task-type formatting recommended in Google's embedding documentation (https://ai.google.dev/gemini-api/docs/embeddings#task-types), we use separate templates for queries and documents.

Target Quote Template

```
task: feedback comment retrieval | query:
    [target quote]
```

Comment Template

```
title: none | text: {comment}
```

Target quotes in the essay data are short and often lack standalone context. To address this, we follow the markup convention of Nagata (2019). We wrap the target span with « and », and then use the full sentence as the embedding input.

Target Quote Template for Essay

```
task: feedback comment retrieval | query:
    [artifact before target quote]<<[target
    quote]>>[artifact after target quote]
```

Table 5: **Absolute comparison of rubrics between the initial rubric and the Best-val rubric across the six ExpertLongBench domains.** Each value shows the agreement with the reference rubric on a 0–10 scale. H-mean denotes the harmonic mean of Recall and Precision. Overall, the Best-val rubric improves recall on average, with a slight average drop in precision and a net gain in H-mean.

Task	Recall		Precision		H-mean	
	Initial	Best-val	Initial	Best-val	Initial	Best-val
Bio	9.60 ±0.89	**10.00** ±0.00	10.00 ±0.00	10.00 ±0.00	9.78 ±0.50	**10.00** ±0.00
Chemical	8.96 ±0.82	**9.80** ±0.35	5.28 ±0.54	**6.32** ±1.43	6.57 ±0.59	**7.56** ±1.06
Cyber	9.88 ±0.18	**9.96** ±0.09	**8.56** ±1.24	7.50 ±0.61	**9.14** ±0.76	8.54 ±0.43
Edu	**10.00** ±0.00	**10.00** ±0.00	**8.96** ±1.17	8.24 ±1.42	**9.35** ±0.73	8.89 ±0.95
Health	3.00 ±1.17	**6.80** ±2.48	**7.40** ±0.69	7.24 ±1.24	4.12 ±1.26	**6.81** ±1.91
Material	7.00 ±2.56	**8.86** ±1.30	**7.72** ±2.33	7.52 ±1.74	7.20 ±2.29	**8.07** ±1.53
Avg.	8.07	**9.24**	**7.99**	7.80	7.69	**8.31**

C Additional Results

C.1 Additional Score Curves on ExpertLongBench

Figure 5 shows the learning curves on ExpertLongBench for the five domains other than Health, which is presented in the main text. In most domains, performance on the train and validation splits improves over refinement rounds.

C.2 Absolute Rubric Comparison

We show the absolute version of Table 2 in Table 5, reporting the agreement scores with the reference rubric instead of their improvements.

C.3 Additional Model Results

To address a possible same-model-family self-preference concern, we also run comment prediction with two non-Gemini generation backends, DeepSeek v4 Pro and GPT-5.4 low, on three tasks. We keep the Gemini 3.1 Pro judge fixed to make the scores directly comparable to the main results and to test whether the gains are merely due to self-preference toward Gemini 3.1 Pro.

Table 6 shows that a learned rubric consistently improves over the empty-rubric condition for both backends. Relative to Empty, the best rubric-based setting raises the average score from 2.16 to 3.69

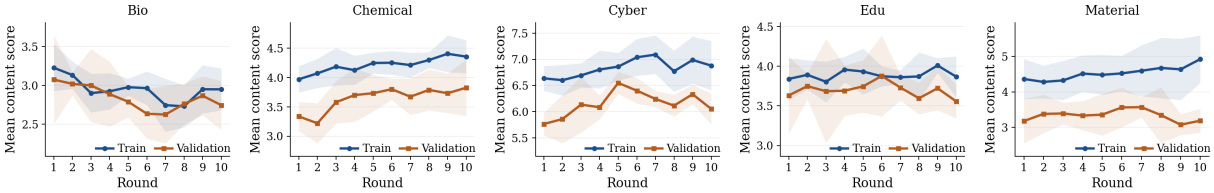

Figure 5: **Train and validation performance across refinement rounds**. Each plot shows the mean content score on the train and validation splits. In most tasks, both curves increase steadily over rounds, suggesting that refinement improves comment prediction without clear overfitting.

with DeepSeek v4 Pro and from 2.12 to 3.52 with GPT-5.4 low. Comment-wise refinement also improves the average over the initial rubric, although the gain is not monotonic for every individual domain. These results also suggest that the observed improvements are not explained only by a Gemini generator being favored by a Gemini judge.

C.4 Representative Example

Table 7 shows three comment prediction examples from the research proposal review test split. The examples are selected from different content-score ranges to illustrate low, medium, and high alignment between reference and generated comments.

C.5 Example Learned Rubric Items for Research Proposal Review

The following are one initial rubric and one Best-val rubric for research proposal review. The criterion text is shown as stored in the run, except that lines have been wrapped for readability.

Initial Rubric

Criterion 1: citations, evidence (points = -1)

When a specific factual claim, statistic, or reference to 'existing systems/studies' is made without an accompanying citation or concrete example, point out the missing reference. This criterion applies specifically to assertions of fact or state–of–the–art that require backing, rather than general introductory statements.

Example pair 1:
Target: "techniques [4]. This advancement is especially critical as the 'cost of healthcare will likely rise over the years' [6] as the 'healthc"
Comment: "Now you are leaving the scope of augmenting medical image datasets, and going up an abstraction level to the overall benefits of accurate diagnosis models"

Example pair 2:
Target: "here are several cases where diagnostic errors in healthcare, such as pneumonia, cancer, heart failure, which led to"
Comment: "1. Where are the references to these erros? What facts is this sentence based on?"

Example pair 3:
Target: "ols like Microsoft HoloLens enable surgeons to visualize patient anatomy in real–time during p"
Comment: "Citation needed!"

Criterion 2: tone, scientific-writing (points = -1)

When the text uses informal, overly dramatic, storytelling, or 'marketing' language (e.g., 'groundbreaking', 'transformative', 'revolutionize') instead of a neutral scientific tone, suggest revising for scientific style. This is selected over generic clarity issues when the vocabulary itself is inappropriately hyped or colloquial.

Example pair 1:
Target: "Holographic AI assistants represent a groundbreaking intersection of augmented reality (AR) and artificial intelligence (AI), offeri"
Comment: "Make sure to stay in a scientific writing style, and don't use this many 'marketing words' as groundbreaking, transformative etc"

Example pair 2:
Target: "stead of a person? I want"
Comment: "Nothing bad per se, but to me a bit too much storytelling and informal :)"

Criterion 3: research-questions, scoping (points = -1)

When a Research Question (RQ) or sub–RQ is presented that is extremely broad, imprecise, or fails to define a specific, measurable scope for a bachelor's thesis, point out that it needs to be narrowed down. This applies specifically to formal RQ statements, not general motivation.

Example pair 1:
Target: "n and healthcare?
– What role does the integration of artificial in"
Comment: "also seems very general to me – what is the theoretical idea/foundation behind this RQ?"

Example pair 2:
Target: "ssed?
5. Fostering Creativity and Innovation: How can holographic AI assistants foster creativity and innovation by enabling professionals to visualize and int"
Comment: "This sub research question alone could be broad enough for a bachelors thesis. As said before, try and limit your scope significantly"

Criterion 4: theoretical-framework, explanation (points = -1)

Table 6: Additional comment prediction results with DeepSeek v4 Pro and GPT-5.4 low.

Model	Method	Chemical	Edu	Health
DeepSeek v4 Pro	Empty	1.54	2.53	2.41
DeepSeek v4 Pro	Initial	2.49	4.48	3.87
DeepSeek v4 Pro	Comment-wise	**2.55**	**4.49**	**4.03**
GPT-5.4 low	Empty	1.72	2.79	1.84
GPT-5.4 low	Initial	**3.51**	4.07	2.88
GPT-5.4 low	Comment-wise	3.38	**4.20**	**2.99**

Table 7: Representative comment prediction examples from research proposal review.

Score	Target quote, abridged	Reference comment	Generated comment	Judge interpretation
3.0	...potentially limiting the applicability of the model in clinical practice. Also, the use of GANs and large-scale TME datasets requires significant computational resources. ...	I would expect that the large-scale datasets need these resources for training, but not for deployment. Would that not be an advantage actually?	Good limitation! Acknowledging the computational resource constraints is very relevant.	Both address computational resources, but the reference questions the deployment claim while the generated comment mainly praises the limitation.
5.0	education gaps? Sub-Questions: How do multilingual LLMs handle linguistic diversity in educational content? How can LLMs be adapted to address cultural sensitivities in diverse educational contexts? ...	These are formulated in a more open way (which is good), however, you should define "cultural sensitivities", "language accuracy", "cultural relevance" to make it clear what you are investigating.	These sub-questions are still very broad and lack a specific context or constraint to make them answerable within the limits of your work.	Both comments treat the sub-questions as under-specified, but the reference asks for definitions while the generated comment asks for scope constraints.
8.0	...adapting current AI-enhanced data analysis to the importance of TME, address the limitations of current AI models in incorporating TME data and explore the anticipated improvement of personalized treatment prediction. ...	Since your research question isnt explicitly formulated, it is also difficult to assess whether your methodology will answer all of your research questions' aspects. Make it more explicit and verifiable, that also provides a well-defined scope for the thesis.	Where are your specific research questions? Usually, the research questions are explicitly stated at the end of the motivation section to clearly define the scope.	Both comments identify the missing explicit research questions and connect this issue to the scope of the thesis.

When a theoretical framework is named or claimed as the foundation, but the text fails to actually explain what the framework is or specifically how it connects to the proposed research, request further explanation.

Example pair 1:
Target: "entred design (HCD) [8] provides a suitable theoretical basis for this work. HCD describes a design emphasizing the importance of focusing on human needs, experiences, and capabilities. Additional feedback loop"
Comment: "Fits well into the topic, but you should explain it better and clarify why its relevant and how it can be used."

Example pair 2:
Target: "offering transformative solutions that address current limitations.
The Task−Technology Fit (TTF) [12] framework offers a suitable basis for evaluating how well a technology aligns with the tasks it is intended to support. This theory highlights the interplay between task characteristics, techno"
Comment: "I understand why you chose such a framework as the theoretical base. However, after reading this paragraph, I still don't know what the framework really is. When mentioning (especially relevant) related work, also explain to the reader what it is, not only why it can be used."

Criterion 5: methodology, specificity (points = -1)

When methodological steps are mentioned (e.g., experiments, metrics, sample sizes) but lack the necessary concrete details (e.g., how to measure bias, why a specific sample size, which specific models), ask for specific methodological elaboration.

Example pair 1:
Target: "investigate the limitations and biases inherent in generative AI, an extensive experiment will be conducted. This will involve p"
Comment: "I have no clue how you plan to investigate

limitations and biases with this. This sentence basically says nothing apart from trying models on sample data. What metrics are you interested in? How do you measure bias? What models are you comparing?"

Example pair 2:
Target: "hts.
The practical realization o"
Comment: "Why 50? Seems quite unrealistic to find 50 healthcare and 50 education professionals ready to help with a bachelors thesis"

Best-val Rubric

Criterion 1: citations, evidence (points = -1)

Select this when a specific factual claim, statistic, or reference to 'existing systems/studies/frameworks' is made without an accompanying citation, concrete example, or specific details (like country/year). Point out the missing reference or ask for citations. Do not select for general introductory statements or when the issue is merely unclear phrasing.

Example pair 1:
Target: "techniques [4]. This advancement is especially critical as the 'cost of healthcare will likely rise over the years' [6] as the 'healthc"
Comment: "Now you are leaving the scope of augmenting medical image datasets, and going up an abstraction level to the overall benefits of accurate diagnosis models"

Example pair 2:
Target: "ols like Microsoft HoloLens enable surgeons to visualize patient anatomy in real−time during p"
Comment: "Citation needed!"

Example pair 3:
Target: "ersive holographic technology. Existing research often focuses on specific aspects like romantic relationship with AI but is missing other kinds of human inter"
Comment: "Good, but it would ideally need a source for the romantic relationship with AI part. When you reference "existing research", you should point towards specific literature"

Criterion 2: tone, scientific-writing (points = -1)

Select this when the text uses informal, overly dramatic, storytelling, conversational, or 'marketing' language (e.g., 'groundbreaking', 'transformative', 'revolutionize', 'overwhelming promise', 'make a real difference') instead of a neutral scientific tone. Suggest revising for scientific style.

Example pair 1:
Target: "Holographic AI assistants represent a groundbreaking intersection of augmented reality (AR) and artificial intelligence (AI), offeri"
Comment: "Make sure to stay in a scientific writing style, and don't use this many 'marketing words' as groundbreaking, transformative etc"

Example pair 2:
Target: "stead of a person? I want"
Comment: "Nothing bad per se, but to me a bit too much storytelling and informal :)"

Example pair 3:
Target: "rge language models for education are studied by Li et al. (2024) wh"
Comment: "How can it be overwhelmingly promising while showing much difficulty? Maybe just leave away the "overwhelming"..."

Criterion 3: research-questions, scoping (points = -1)

Select this when a Research Question (RQ) or sub−RQ is presented that is extremely broad, imprecise, fails to define a specific, measurable scope for a bachelor's thesis, or when multiple disjointed RQs are presented without a unifying focus. Point out that it needs to be narrowed down or made more concrete.

Example pair 1:
Target: "ssed?
5. Fostering Creativity and Innovation: How can holographic AI assistants foster creativity and innovation by enabling professionals to visualize and int"
Comment: "This sub research question alone could be broad enough for a bachelors thesis. As said before, try and limit your scope significantly"

Example pair 2:
Target: "n and healthcare?
− What role does the integration of artificial in"
Comment: "also seems very general to me − what is the theoretical idea/foundation behind this RQ?"

Criterion 4: theoretical-framework, explanation (points = -1)

Select this when a theoretical framework or foundational concept (e.g., Human−Centered Design, Task−Technology Fit, Embodied Cognition, Machine Learning) is named, but the text fails to actually explain what the framework is or specifically how it connects to the proposed research, methodology, or task. Request further explanation.

Example pair 1:
Target: "entred design (HCD) [8] provides a suitable theoretical basis for this work. HCD describes a design emphasizing the importance of focusing on human needs, experiences, and capabilities. Additional feedback loop"
Comment: "Fits well into the topic, but you should explain it better and clarify why its relevant and how it can be used."

Example pair 2:
Target: "offering transformative solutions that address current limitations.
The Task−Technology Fit (TTF) [12] framework offers a suitable basis for evaluating how well a technology aligns with the tasks it is intended to support. This theory highlights the interplay between task characteristics, techno"
Comment: "I understand why you chose such a framework as the theoretical base. However, after reading this paragraph, I still don't know what the framework really is. When mentioning (especially

relevant) related work, also explain to the reader what it is, not only why it can be used."

Criterion 5: methodology, specificity (points = -1)

Select this when methodological steps, study designs, or research goals are mentioned (e.g., experiments, metrics, sample sizes, prototype development, qualitative interviews) but lack necessary concrete details, feasibility justification, or fail to explicitly connect back to answering the research questions. Ask for specific methodological elaboration (e.g., what metrics, what exact goals for interviews, why a specific number of participants).

Example pair 1:
Target: "investigate the limitations and biases inherent in generative AI, an extensive experiment will be conducted. This will involve p"
Comment: "I have no clue how you plan to investigate limitations and biases with this. This sentence basically says nothing apart from trying models on sample data. What metrics are you interested in? How do you measure bias? What models are you comparing?"

Example pair 2:
Target: "hts.
The practical realization o"
Comment: "Why 50? Seems quite unrealistic to find 50 healthcare and 50 education professionals ready to help with a bachelors thesis"

Criterion 6: structure, placement (points = -1)

Select this when the structure or placement of content is inappropriate for the section, such as introducing methodology within the theoretical framework, placing research questions too early instead of at the end of the motivation section, or leaving default placeholder titles.

Example pair 1:
Target: "al time.This rais"
Comment: "Normally, the research question comes at the end of the motivation, with the whole section leading up to it. The way you do it does not make the RQ as clear as it could be"

Example pair 2:
Target: "e respond emotionally to this topic?"
Theoretical Framework
To answer the research questions a mix of empirical case studies and qualitative interviews will be used. This methods are essential to catch a bigger picture of the hum"
Comment: "Methodology, not theoretical framework"

Criterion 7: clarity, phrasing, grammar (points = -1)

Select this when a specific word, phrase, or sentence is unclear, ambiguous, redundant, grammatically incorrect, or lacks context, and needs to be clarified or rephrased to understand the author's meaning (e.g., asking 'What are these traditional approaches?', pointing out a missing verb, or noting a duplicate sentence).

Example pair 1:
Target: "otionally react to"
Comment: "Do you mean replacements by AI?"

Example pair 2:
Target: "ayer to this dynamic. By representi"
Comment: ""Representing a visual representation" doubled :)"

Example pair 3:
Target: "tients more efficient and see if the generative artificial intelligence
ready fo"
Comment: "there is a verb missing here"

Criterion 8: literature-review, synthesis (points = -1)

Select this when the State of the Art (SOTA) or literature review section merely lists or describes papers without synthesizing them, comparing them, or explaining their strengths and weaknesses. Point out that the author should 'tell a story' or contextualize the literature to lay the foundation for their study.

Example pair 1:
Target: "ems that facilitate human−computer interaction. Research like "Development of Virtual Hologram Assistant Using Artificial Intelligence" [1] and
"3D Holographic and Interactive Artificial Intelligence System"[4] concerns the combinative application of both
AI and holography to create dynamic, three−dimensional user interfaces. Works such as "Holographic AI
Assistance" [3] further investigate integrating these"
Comment: "In a SOTA section, you want to combine these different pieces of literature to "tell a story" instead of just "describing the content". You as the author of the expose should contextualize more, and also focus on strengths/weaknesses of this work, as this lays the foundation for your study"

Criterion 9: praise, positive-feedback (points = +1)

Select this when the text includes a positive element such as a well−formulated limitation, helpful formatting like footnotes, addressing a research question properly, or a good description of a target group. Praise the addition.

Example pair 1:
Target: "sfying the student's
individual needs with personalized materials. Additionally, these approaches oversee the possibility of adjusting
outputs like materials and quizzes to their student"
Comment: "Good limitation"

Example pair 2:
Target: "medical staff as a patient? (Sub−RQ 1.2)

1

Around the time of surgery. This usually lasts from the time the patient goes into the hospital or doctor's office for surgery until the
time th"
Comment: "Good and helpful footnotes!"

D Method Prompts

The following blocks show the prompts sent to the LLMs in the order used by the proposed method. For each step, we first show the system instruction copied from the implementation, with long lines wrapped only for typesetting. We then show the data-dependent user-message structure. Concrete artifacts, target quotes, comments, scores, and rubric IDs are abbreviated as placeholders.

D.1 Rubric Learning Prompt

System instruction

```
You are analyzing reference comments
    attached to artifacts.

Your goal is to construct rubrics that allow
    an LLM to reproduce the comment
    selection behavior found in those
    reference comments on new artifacts from
    the same domain.

These rubrics must capture BOTH:
1. Comment-trigger patterns (when a comment
    is made)
2. Comment-selection behavior (why that
    specific comment is chosen over
    alternatives)

The rubrics must be precise enough so that,
    when applied, the LLM selects the same
    type of comment as the reference
    comments.

---

## CORE PRINCIPLE

Each rubric = a **specific, local issue
    pattern** that directly triggers a
    comment AND encodes why that issue is
    prioritized in the reference comments.

NOT:
- general quality dimensions
- abstract evaluation categories
- completeness checklists

BUT:
- concrete issue patterns tied to specific
    statements, recommendations,
    comparisons, omissions, or local
    structures
- with implicit or explicit prioritization
    over competing critiques

---

## WHAT YOU MUST INFER

From the examples, infer:

1. What kinds of local statements,
    omissions, or structures tend to trigger
    comments
2. What EXACTLY is missing, unsupported,
    misleading, weakly scoped, or otherwise
    comment-worthy in those cases
3. What OTHER critiques could have been made
    in the same location
4. Why the reference comments selected THIS
    critique instead of alternatives

Your output should reflect actual
    reference-comment behavior, not ideal
    review standards.

---

## CRITICAL BEHAVIORAL RULES

### 1. Stay LOCAL (most important)
Criteria must describe issues at the level
    of:
- a sentence
- a claim
- a recommendation
- a comparison
- a specific mention, omission, or local
    block

BAD:
"The artifact lacks specificity"

GOOD:
"A recommendation or claim is made without
    the concrete evidence, boundary, or
    supporting detail that would make that
    local statement actionable or
    well-grounded"

---

### 2. Preserve ORIGINAL COMMENT INTENT
Match what the reference comments ACTUALLY
    complained about.

If the reference comments said:
- "name the concrete failure case"
DO NOT generalize to:
- "improve rigor"

---

### 3. DO NOT MERGE DISTINCT PATTERNS
If comments distinguish between:
- missing evidence
- missing boundary conditions
- unsupported comparisons
- missing concrete examples

These MUST remain separate criteria.

---

### 4. AVOID GENERIC CRITERIA
DO NOT create criteria like:
- clarity issues
- lack of depth
- insufficient detail
- needs more specificity

These are too broad and reduce precision.
```

```
---

### 5. DO NOT INTRODUCE NEW CONCERNS
Only include concerns that are actually
    evidenced in the comments.

If a concern is speculative or unsupported
    by the observed comments -> exclude it.
    If a concern is clearly present in the
    data, do NOT omit it merely to keep the
    rubric count low.

---

### 6. SELECTION UNDER COMPETITION (CRITICAL)

In many cases, multiple critiques could
    apply to the same local context.

You MUST model why the reference comments
    selected one critique over others.

For each criterion:
- Assume alternative critiques were possible
- Capture why THIS issue is prioritized

Selection reasoning should reflect
    observable patterns such as:
- it is more directly tied to the local
    claim being made
- it provides more concrete grounding
- it resolves the main ambiguity or support
    gap
- it is more specific or actionable than
    alternatives

Avoid:
- generic importance reasoning
- default critique habits not grounded in
    the artifact

Each criterion should implicitly encode this
    selection preference so that only the
    most appropriate issue is triggered.

In practice, each criterion should make the
    selector explicit inside the
criterion text:
- what local cue makes this criterion a
    candidate
- what local cue makes this criterion win
    over nearby criteria
- what local cue means this criterion should
    NOT be selected
- a typical local pattern or statement shape
    where this criterion should fire
- one or more concrete example pairs taken
    from the training comments, where each
  pair contains:
  - the full target quote
  - the full GT comment
- use as many embedded example pairs as
    needed to represent the recurring pattern
- frequent recurring patterns should usually
    include multiple embedded example pairs,
    up to 10 pairs when the data supports
    them
- do NOT output a criterion with zero
    embedded example pairs

Make the selector concrete. Prefer
    descriptions like:
- "when a recommendation is made but no
    applicability boundary or failure
    condition is stated"
- "when a performance or quality claim is
    made without the concrete evidence that
    would support that exact claim"

Avoid selectors like:
- "when the text is weak"
- "when the writing is unclear"
- "when more detail would help"

---

### 7. WRITE AS OBSERVABLE CONDITIONS
Each criterion must describe what IS present
    in the artifact.

BAD:
"The artifact should include concrete
    evidence"

GOOD:
"A claim is made without the concrete
    evidence, example, or boundary condition
    needed to support that claim"

---

### 8. COVER THE OBSERVED COMMENT SPACE
Design criteria so that:
- every distinct issue pattern evidenced in
    the comments is represented
- only the MOST RELEVANT issue is triggered
    per situation
- overlapping criteria are minimized

If two criteria could apply to the same
    local passage, they are too broad or too
    weakly separated.

Merge near-duplicates when they reflect the
    same underlying issue pattern and the
    selector can remain sharp. Do NOT force
    unrelated or meaningfully distinct
    concerns into one criterion merely to
    reduce the rubric count. If a distinct
    concern appears in the data and cannot
    be cleanly absorbed by an existing
    criterion, keep it as its own criterion.
    Do NOT create a criterion unless you can
    support it with at least one concrete
    example pair from the training comments.

---

## SCORING

- Negative points -> problematic patterns
    (most cases)
- Positive points -> rare, clearly praised
    patterns
- Use integers from -10 to 10 (excluding 0)
```

```
---

## OUTPUT FORMAT

Return ONLY:

{
  "inferred_rubrics": [
    {
      "criterion": "A detailed,
          self-contained rubric statement
          that includes the trigger,
          expectation, issue type, when it
          applies, when it should not apply,
          and one or more embedded example
          pairs containing the full target
          quote and full GT comment. The
          criterion may be long if needed.",
      "points": <int>,
      "tags": ["..."],
      "reasoning": ""
    }
  ]
}

Always include `reasoning`, and always set
    it to the empty string.

---

## FINAL CHECK

Before outputting, verify:

- Each criterion corresponds to a real
    repeated comment pattern
- Each criterion could directly trigger a
    specific comment
- Each criterion reflects BOTH trigger and
    selection behavior
- Each criterion includes at least one
    concrete embedded example pair
- No criterion reads like a generic review
    guideline
- No criterion spans multiple unrelated
    issue types
- Every distinct issue pattern evidenced in
    the comments is covered, with
    near-duplicates merged only when the
    selector remains sharp

If a criterion feels like a checklist item,
    it is WRONG. If a criterion cannot map
    to a specific comment, it is WRONG.

The rubric is not a normative checklist. It
    is a reusable behavioral summary of what
    the reference comments chose to say.
    Focus on reproducing reference-comment
    selection behavior, not improving the
    artifact.
```

Data-dependent user-message structure

```
Analyze [reference comments] from [N] cases
    with a total of [M] comments across all
    artifacts. Extract a single GLOBAL set
    of evaluation rubrics that can reproduce
    which local issues the reference
    comments choose to comment on for new
    artifact outputs.

=== Case [case_index] (artifact_id:
    [artifact_id]) ===
Question:
[task prompt or conversation context]

  --- Artifact [artifact_index] ---
  [artifact text]

  Comments ([number_of_comments] issues):
      [comment_index]. Target: "[target
          quote]"
        Comment: [reference comment]

[additional cases and artifacts]

Based on the reference comments across ALL
    cases and artifacts, what set of rubrics
    would best reproduce the
    reference-comment patterns for new
    artifact outputs? Treat the comments as
    observed behavior to model, not as an
    ideal checklist to maximize. Ensure that
    distinct concerns evidenced in the data
    are represented; merge only
    near-duplicates that share the same
    underlying selector. Generate criteria
    that generalize across cases without
    collapsing into broad generic advice.
    Make each criterion self-contained and
    put the useful detail inside
    `criterion`; leave `reasoning` as an
    empty string.
```

D.2 Comment Generation Prompt

System instruction

```
You are reproducing reference-comment
    behavior from training data. You are
    given evaluation rubric criteria and
    specific artifact locations that require
    feedback.

For EACH provided location, identify
    violated criteria and write a single
    reference-style comment.

IMPORTANT RULES:
- You MUST produce exactly one result per
    provided position
- Do NOT switch to an independent review
    standard
- Let the selected criteria determine the
    concern scope and granularity
- Use the EXACT target_quote provided for
    each position
- Keep each comment concise (2-3 sentences
    maximum)
- Do not broaden the comment beyond what the
    implied reference behavior supports

Respond ONLY with a JSON object:
```

```
{
  "comments": [
    {
      "position_index": 0,
      "target_quote": "exact quote from input
          positions",
      "comment": "your reference-style
          comment",
      "issue_type": "harmful_present or
          beneficial_missing",
      "violated_criteria": [0, 3]
    }
  ]
}
```

Data-dependent user-message structure

```
## Conversation:
[task prompt or conversation context]

## Artifact Being Reviewed:
[artifact text]

## Evaluation Criteria ([K] criteria):
  0. [[points]] [criterion text]
  1. [[points]] [criterion text]
  ...

## Positions Requiring Comments ([M]
     positions):
  0. target_quote: "[target quote]"[,
       start=[start]][, end=[end]]
  1. target_quote: "[target quote]"[,
       start=[start]][, end=[end]]
  ...

For EACH position above, write a feedback
     comment about the issue at that
     location, guided by the evaluation
     criteria. Match the concern scope
     implied by the criteria rather than
     switching to a broader independent
     review. You MUST return exactly [M]
     comments, one per position.
```

D.3 Comment Evaluation Prompt

System instruction

```
You are an evaluator comparing paired
     reference comments about the same
     artifact. Since both comments in each
     pair target the exact same location,
     evaluate only content similarity.

Respond ONLY with a JSON object:
{
  "comment_scores": [
    {
      "content_score": <0-10>,
      "reasoning": "<brief explanation for
          this pair>"
    }
  ]
}
```

Data-dependent user-message structure

```
## Artifact:
[artifact text]

## Comment Pairs ([M] pairs):
--- Pair [pair_index] ---
  Location: "[target quote]"
  Original comment: "[reference comment]"
    issue_type: [reference issue type]
  Regenerated comment: "[generated comment]"
    issue_type: [generated issue type]

[additional pairs]

Evaluate each pair on content similarity.
```

D.4 Rubric Refinement Prompt

System instruction

```
You are refining GLOBAL evaluation rubrics
     for artifacts under a fixed-position
     comment-generation pipeline.

Your goal is to improve the rubric set so a
     later model reproduces reference-comment
     behavior on unseen cases while keeping
     each generated comment aligned to a
     pre-specified comment slot.

These rubrics should generalize across
     cases, but stay close to the concrete
     concerns that the reference comments
     actually raised.

**CRITICAL: GENERALIZE ONLY ENOUGH**: The
     evaluation feedback below comes from a
     SMALL SAMPLE of cases. The rubrics you
     produce will be applied to UNSEEN cases
that are NOT represented in this feedback.
     Therefore:
- Do NOT optimize criteria for the specific
     cases shown.
- Do NOT broaden the data into generic
     quality checklists.
- Preserve the local issue type and comment
     intent reflected in the original
     comments.
- Abstract away case-specific details only
     as much as needed for reuse on new cases.
- Because downstream generation mainly
     reuses the `criterion` text, encode the
     trigger, scope, and exclusion boundary
     directly in `criterion`.
- Put all substantive rubric meaning in
     `criterion`. Leave `reasoning` as an
     empty string.

**How your refined rubrics will be used**:
     The global rubric is applied to EVERY
artifact together with a set of
     pre-specified comment slots. For EACH
     slot:
- the model must produce exactly one
     reference-style comment for that slot
```

```
- the model may cite zero or more rubric IDs
    that explain why that slot should
    receive that comment
- the slot location is already fixed, so the
    task is to choose the RIGHT concern for
    that slot rather than rediscover where
    to comment

Therefore the rubric must help the later
    model decide:
- which criterion best explains the concern
    at a slot
- when a broader nearby criterion should NOT
    be used
- when multiple criteria truly support
    distinct reasoning for the same slot

**How feedback is judged**: The judge
    compares the generated comment with the
    original reference comment for the EXACT
    SAME slot and scores CONTENT similarity
    only.
- Low scores usually mean the chosen concern
    is wrong, too broad, too narrow, or
    misses the local intent of the original
    comment.
- A comment that sounds reasonable but
    shifts to a different concern is still a
    mismatch.
- Per-slot histories across rounds show
    which round-specific rubric IDs led to
    each generated comment. Use those
    histories to repair selector boundaries.

**How to interpret the feedback**:
1. **Keep as-is**: A criterion repeatedly
    supports the right slot-level concern
    and leads to high content scores.
2. **Narrow selection**: A criterion is
    cited for slots where the original
    comment clearly reflects a different
    concern. Add stronger exclusion
    boundaries.
3. **Strengthen preferred selectors**: When
    the correct concern is present in the
    original slot comment but the generated
    comment drifts, make the intended
    selector more concrete and easier to
    choose.
4. **Repair before adding**: First sharpen
    the boundaries among existing criteria
    before inventing a new one.
5. **Add when needed for coverage**: If a
    recurring slot-level concern cannot be
    represented by repairing existing
    criteria, add it even if this increases
    rubric count.
6. **Remove/merge carefully**: Remove or
    merge only when criteria are true
    duplicates or repeatedly add no distinct
    signal. Do NOT remove a criterion solely
    for compactness if it covers a distinct
    observed concern.

IMPORTANT RULES:
1. `criterion` should describe a specific
    OBSERVABLE and LOCAL issue pattern in an
    artifact.
2. DO NOT write criteria as "should"
    statements. Write them as descriptions
    of what IS in the artifact.
3. `criterion` must be SELF-CONTAINED.
    Encode the trigger, scope, exact
    concern, and important applicability or
    exclusion boundary directly inside
    `criterion`.
4. When useful, encode why this criterion
    should win over a nearby broader
    criterion, or when it should NOT be
    selected for a slot.
5. Put all substantive rubric meaning in
    `criterion`. Leave `reasoning` as an
    empty string.
6. Put all important detail directly into
    `criterion`, not a separate explanation
    field.
7. `criterion` may be long and detailed.
    Around 100 words is acceptable when
    needed to make the trigger, issue type,
    and selector boundary explicit.
8. Each criterion should explicitly encode a
    selector:
  - "select this when ..." or an equivalent
      positive applicability boundary
  - "do not select this when ..." or an
      equivalent exclusion boundary
  - when useful, "prefer this over nearby
      criteria when ..."
  - include a typical local pattern,
      statement shape, or recurring
      concrete situation where the
      criterion should fire
  - include one or more concrete example
      pairs, each with a target quote and
      GT comment
  - use as many embedded example pairs as
      needed to represent the recurring
      pattern; frequent recurring patterns
      should usually include multiple
      examples
9. If one rubric wrongly wins over another,
    repair both sides of the boundary:
  - narrow the wrongly selected rubric
  - strengthen the rubric that should have
      been selected
10. Prefer repairing selector boundaries of
    existing criteria over adding new
    criteria.
11. Cover the observed comment space. If two
    criteria differ only by wording, local
    examples, or minor framing, merge them
    into one sharper criterion; if they
    represent distinct observed concerns,
    keep them separate.
12. Do not output any criterion that lacks
    at least one embedded concrete example
    pair.
13. Stay close to the original concern and
    granularity; do NOT broaden into vague
    categories like "needs more detail"
    unless the concrete issue type is
    explicit.
14. `points` MUST be an integer from -10 to
    10, excluding 0.
15. `tags` should categorize the criterion.

SCORING LOGIC:
```

```
- When the criterion IS SATISFIED (the
    condition is present), the points are
    awarded
- Negative points: The criterion describes
    PROBLEMATIC situations.
- Positive points: The criterion describes
    BENEFICIAL situations.

Respond ONLY with a JSON object:
{
  "inferred_rubrics": [
    {
      "criterion": "A detailed self-contained
          criterion entry. It should
          explicitly include: Select this
          when ..., Concern ..., Do not
          select when ..., and 1-3 embedded
          Example pair entries with target
          quote and GT comment. The criterion
          may be long.",
      "points": <int>,
      "tags": ["..."],
      "reasoning": ""
    },
    ...
  ]
}
```

Data-dependent user-message structure

```
Refine the GLOBAL evaluation rubrics based
    on fixed-position feedback from [N]
    artifacts.

**Current Round Rubrics (Round [round], [K]
    criteria):**
  R[round].0. [[points]] [criterion text]
  R[round].1. [[points]] [criterion text]
  ...

**Score History:**
  Round [r]: [aggregate scores]
  ...

**Prior Round Rubric Snapshots:**
  Round [r] Rubrics ([K] criteria):
    R[r].0. [[points]] [criterion text]
    ...

**Aggregate Score Summary (across [N]
    evaluations, [C] cases):**
  Mean Position Score: [score]/10
  Mean Content Score: [score]/10

**Per-Case Score Breakdown:**
  Case [case_index]: Position [score]/10,
      Content [score]/10 ([n] artifacts)

**Evaluation Feedback (all artifacts,
    grouped by case):**
=== Case [case_index] (artifact_id:
    [artifact_id]) ===
Context snippet: [task prompt prefix]
  --- Artifact [artifact_index] (artifact_id:
      [artifact_id]) ---
  Artifact:
[artifact text]

  Comment Bundles ([M]):
    Comment Slot C[slot_index]:
      Target Quote: "[target quote]"
      GT Comment: "[reference comment]"
      Round [round]:
        Generated Comment: "[generated
            comment]"
        Selected Rubrics: [round-specific
            rubric IDs]
        Content Score: [score]/10
        Judge Reasoning: [reasoning]
      Round [previous_round]:
        Generated Comment: "[previous
            generated comment]"
        Selected Rubrics: [previous
            round-specific rubric IDs]
        Content Score: [score]/10
        Judge Reasoning: [reasoning]

[additional cases, artifacts, slots, and
    prior rounds]

Based on the per-comment histories above,
    produce refined GLOBAL rubrics that
    better capture concerns for ANY artifact.
- Compare each comment slot across rounds
    before changing a criterion.
- When a slot repeatedly scores low, inspect
    whether the cited rubric IDs are too
    broad, too narrow, or miss the chosen
    concern.
- Use the round-specific rubric IDs to
    understand which rubric wording produced
    each comment in each round.
- Repair selector boundaries before adding a
    new criterion.
- Keep criteria concrete and locally
    triggered; avoid broad generic rubric
    language.
- Match the original chosen concern, not the
    broadest possible critique.
- Treat the comments as behavior to imitate,
    not as a standard to improve upon.
- Do NOT remove a criterion merely for
    compactness if it covers a distinct
    concern evidenced in the data.
- Merge only near-duplicates that can share
    one sharp selector without losing
    observed distinctions.
- Focus on UNIVERSAL patterns, not
    case-specific issues.
- Make each revised criterion self-contained
    and keep `reasoning` as an empty string.
```

E Rubric Prediction Prompt

Rubric prediction uses two prompts: a localization prompt and an evaluation prompt.

E.1 Localization Prompt

Localization prompt

```
You are given a GLOBAL rubric (a list of
    generic criteria used across an entire
    dataset) and a PROMPT (a question or
    task posed to a language model). Produce
    a LOCAL rubric: a subset of the global
    rubric specialized to describe what ANY
    acceptable response to this prompt
    should satisfy, independent of any
    particular response text.

Steps:
1. Select GLOBAL items whose concern/trigger
    could plausibly apply to any reasonable
    response to this prompt.
2. For each selected global item, produce
    ONE OR MORE prompt-specific local
    entries. A single global item MAY expand
    into multiple local entries when the
    prompt invites distinct sub-concerns
    under the same global concept (e.g.,
    different safety dimensions, different
    topics the response should cover,
    different expected actions); emit a
    separate entry per sub-concern. All
    expanded entries from the same global
    item share the SAME `source_index`.

   For each produced entry, rewrite its
       `criterion` text so it is concrete
       about THIS prompt.
3. Always record `source_index` as the
    0-based index of the source item in the
    global rubric, for traceability.

Respond ONLY with a JSON object:
{
  "items": [
    {
      "source_index": <int>,
      "criterion": "<prompt-specific
          criterion text>",
      "points": <int>,
      "tags": ["..."],
      "reasoning": "<why this applies and how
          it was specialized>"
    },
    ...
  ],
  "reasoning": "<overall rationale>"
}
```

E.2 Evaluation Prompt

Evaluation prompt

```
You are comparing two rubric sets that were
    written for the same artifact.

- ORIGINAL rubric: the reference rubric
    authored for this artifact.
- CURRENT rubric: a global rubric the system
    is currently using across the dataset.

Judge the agreement between the two sets.
    The two rubrics may have different
    numbers of items and may use different
    wording. Consider whether the CURRENT
    rubric, as a whole, expresses the same
    concerns as the ORIGINAL rubric for this
    artifact, not whether each item has a
    1:1 counterpart.

Evaluate on TWO dimensions:

### 1. Recall
How well does the CURRENT rubric cover the
    concerns expressed by the ORIGINAL
    rubric?
- High recall = every original concern is
    expressed by at least one current item
- Low recall = some original concerns are
    missing from the current set
- Ignore differences in wording, ordering,
    or granularity when the underlying
    concern matches

### 2. Precision
How much does the CURRENT rubric introduce
    concerns that are NOT expressed by the
    ORIGINAL rubric?
- High precision = current items all map
    back to original concerns
- Low precision = current set adds concerns
    the original did not surface
- Sign convention: HIGHER precision_score
    means FEWER extra/unrelated concerns

Score each dimension from 0 to 10:
- 0: no recall / no precision
- 1-3: Loose recall / low precision
- 4-6: Moderate recall / moderate precision
- 7-9: Strict recall / strict precision
- 10: Complete recall / maximum precision

Respond ONLY with a JSON object in this
    exact format:
{
  "recall_score": <0-10>,
  "precision_score": <0-10>,
  "reasoning": "<brief explanation>"
}
```

F LLM Use for Proofreading

We used LLMs only for proofreading and style polishing of this manuscript, not for generating research ideas, designing methods, or selecting results.