

ReFeedGraph: Graph-Grounded Reusable Feedback for Debate Improvement

Priya Pitre^{*1} Omkar Joshi^{*2} Timon Ziegenbein³ Henning Wachsmuth³

¹Virginia Tech ²North Carolina State University

³Leibniz University Hannover, L3S Research Center

priyapitre@vt.edu ojjosshi@ncsu.edu t.ziegenbein@ai.uni-hannover.de h.wachsmuth@ai.uni-hannover.de

Abstract

Debate feedback is useful only when it explains why a side lost and how that side can revise to become more competitive. However, LLM-generated feedback often sounds polished while remaining generic and vague. We introduce REFEEDGRAPH, a graph-based method that turns prior debate feedback into reusable principles. It collects multi-framework judge feedback, extracts weaknesses and actionable suggestions, abstracts them into principles, and stores them in a graph. For a new debate, REFEEDGRAPH retrieves relevant principles and adapts them into targeted revision feedback. We evaluate the generated feedback by recording whether it improves the weaker side after rewriting. Across three existing datasets (DebateArt, Intelligence Squared, and British Parliamentary debates), REFEEDGRAPH outperforms direct LLM and structured-prompt baselines, producing more specific and actionable feedback that leads to winning more debates. Code: <https://github.com/priyapitre/ReFeedGraph>

1 Introduction

Debate is ubiquitous: it appears in formal competitions, classrooms, civic and political discourse, workplaces, and everyday disagreements (Gutmann and Thompson, 2004; Taylor, 2013; Drury, 2019). In all of these settings, people improve by learning not only what to argue, but also how to respond, compare, and adapt when challenged. This makes *feedback* central to debate training. A debate is not won by having fluent arguments in isolation. It is won by comparing claims, answering the other side, proving that one clash matters more than another, and showing why the judge should prefer one side’s framing of the round (Gordon et al., 2007). As a result, useful debate feedback must explain which argument failed, which response was missing, which comparison was underdeveloped, and

^{*}Equal contribution.

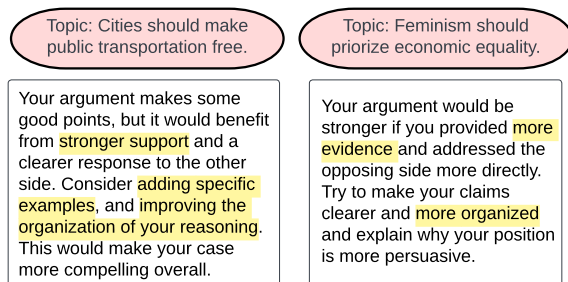


Figure 1: Generic LLM feedback on different topics showing: vague, non-actionable, non-clash specific, polished (but unhelpful) feedback. The feedback on completely different speeches sound identical with generic suggestions.

how the debater should revise the argument in a future round. This makes good debate feedback difficult to generate at scale.

Large Language Models (LLMs) can produce feedback quickly, but their comments are often vague, non-actionable, and insufficient across debates (Rashkin et al., 2025). LLM-generated feedback can be long without being specific, polished without being useful, and similar in tone for vastly different debates. Model responses for most debates often rely on generic phrases such as ‘add evidence’ or ‘improve weighing’ without identifying the relevant clash, showing where evidence is needed, or giving any more nuanced feedback. On the other hand, giving specific, concrete and nuanced human feedback requires time, training, and domain knowledge. Moreover, even when LLM or human feedback is available, there are no universally agreed upon metrics to judge the quality of the feedback.

Recent work evaluates feedback using pre-defined dimensions such as specificity, helpfulness, validity, and actionability (Chu et al., 2026; Thakkar et al., 2025; Qian et al., 2025; Rashkin et al., 2025). Nair et al. (2024) offer a different

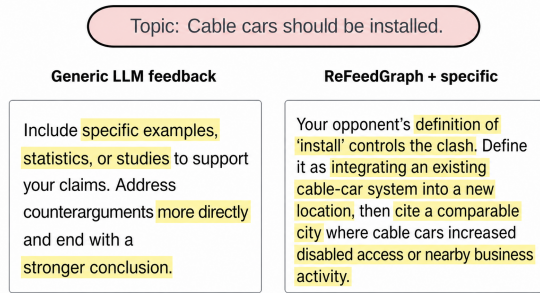


Figure 2: **Generic versus retrieved feedback for the same debate.** Abridged feedback excerpts for the motion “Cable cars should be installed.” Generic LLM feedback provides broadly applicable advice, whereas REFEEDGRAPH+specific identifies the decisive definitional clash and offers a concrete, debate-grounded revision strategy. Yellow highlights emphasize the actionable content of each response.

view of the problem. Instead of defining good feedback through a fixed checklist, their PROF framework evaluates feedback by its effect on revision. They propose that a feedback is useful if a student can use it to produce a better revised essay. However, Nair et al. (2024) leave an important question open:

How to generate feedback that is likely to improve revision in the first place?

In this paper, we address the outlined limitations by presenting a computational approach that makes feedback generation structured, grounded, and reusable. To obtain specific and actionable LLM feedback, we guide multiple judges in an agent setting with debate and argumentation frameworks from theory (Toulmin, 1958; Teich, 1987; Walton et al., 2008; Bengzon et al., 2025; Melamed, 2026), allowing the approach to capture diverse forms of critique such as clash, weighing, warrants, framing, and responsiveness. The resulting feedback is grounded in explicit judging guides rather than generated from an unconstrained prompt.

In a second step, we then abstract debate-specific feedback into general principles: fundamental values of debates that are applicable across debates in certain cases. These store the underlying failure pattern, where a principle applies and how it can be acted on. When stored in a graph form, our nodes connect various failure patterns and their underlying principles to create a comprehensive feedback that one debate or judge would be incapable of doing. When a new debate comes in, the approach retrieves relevant principles from this graph and in-

stantiates them with the specific claims and clashes in the current round.

This allows the system to reuse what has worked in prior debates while still generating feedback that is grounded in the specific claims, clashes, and weaknesses of the current round. Rather than treating each feedback instance as isolated, our approach builds a memory of recurring debate failures and the repair strategies that address them.

Using rewrite-based evaluation, we experiment with REFEEDGRAPH on three existing datasets: DebateArt (Liang et al., 2024), Intelligence Squared (Zhang et al., 2016), and British Parliamentary debates (Liang et al., 2024). Feedback is useful if it helps the weaker side improve after revision, and especially if the revised speech can close or reverse the original winner-loser gap. Across datasets, REFEEDGRAPH produces larger revision gains than direct LLM feedback and structured-prompt baselines, while also generating feedback that is judged more specific and actionable. These results suggest that debate feedback improves when models do not generate advice from scratch, but instead retrieve and adapt reusable principles learned from prior critiques.

Altogether, this paper makes two contributions. First, we introduce REFEEDGRAPH, a graph-based method for converting debate critiques into reusable feedback principles. Second, we evaluate feedback by its downstream effect on revision, showing that reusable graph-grounded feedback can produce more competitive rewritten speeches than standard LLM feedback baselines.

2 Related Work

Feedback Generation Automated feedback generation has been studied extensively in writing assessment and educational NLP, where systems aim to provide revision-oriented comments in addition to scores (Stahl et al., 2024; Misgna et al., 2025; Chu et al., 2026). Prior work on argumentative writing shows that useful feedback often requires identifying claims, premises, argument strength, missing support, and other discourse-level features rather than only surface-level writing quality (Wang et al., 2022; Wambsganss et al., 2021). Recent LLM-based systems extend this direction to large-scale educational assignments and scientific writing, using synthetic datasets, rubric grounding, multi-agent review, or document-focused feedback to improve specificity and usefulness (Qian et al.,

2025; Chamoun et al., 2024; D’Arcy et al., 2024; Thakkar et al., 2025). These works show that feedback can be generated at scale, but they primarily target essays, assignments, or scientific reviews. In contrast, debate feedback must diagnose why a side lost a particular clash and provide a revision that could make that side more competitive.

Reusable Feedback Methods A separate line of work studies how feedback, reflections, or past experiences can be reused across attempts. Retrieval-augmented generation uses external text memory at inference time, while self-refinement methods use feedback to revise model outputs without changing model weights (Lewis et al., 2020; Madaan et al., 2023). Agent-learning methods such as Reflexion and ExpeL store verbal reflections or natural-language insights from past experience and reuse them to improve future decisions (Shinn et al., 2023; Zhao et al., 2024). Other approaches store mistake cases or correction traces so models can avoid repeating earlier errors (Wang and Li, 2023; An et al., 2023).

The closest work to our objective is the PROF framework because it evaluates feedback by whether simulated student revisions improve, rather than by whether the feedback satisfies predefined attributes (Nair et al., 2024). However, PROF does not solve the problems with LLM feedback generation that have been highlighted above. Feedback can still be vague, inconsistent, or strategically misdirected, even if it is later evaluated through revision. Moreover, if a system generates feedback from scratch for every debate, then each evaluated feedback instance remains isolated. The system discards the evidence created by prior evaluations and has no mechanism for carrying forward which feedback patterns helped, which were one-off comments, and which reflected recurring strategic failures. Our work adapts the outcome-based view of Nair et al. (2024) to debate, but it adds a text memory of reusable feedback principles derived from multi-perspective judge comments.

3 Approach

This section presents REFEEDGRAPH, a debate-oriented feedback generation approach. REFEEDGRAPH is retrieval-based rather than fine-tuned, making it explainable, grounded, and resource-efficient through prompting alone. At training time, it constructs a *feedback memory graph* from a set of past debates. To this end, it converts framework-

specific judge comments into reusable principles and stores them in a graph that links concrete debate failures to abstract feedback principles while preserving the source evidence for each principle. At test time, REFEEDGRAPH diagnoses the weaker side in a new debate, retrieves relevant principles from this graph, and adapts them into debate-specific feedback, rewrites the target speech, and evaluates whether the rewrite improves performance.

3.1 Problem Statement

Let $x = \langle m, s_1, s_2, D, y \rangle$ be a one-versus-one debate instance, where m is the motion, s_1 and s_2 are the two opposing sides, D is the full debate transcript, and $y \in \{+, -\}$ is the outcome when available (i.e., which side won the debate). Our goal is to generate feedback f for the weaker side $s^- \in \{s_1, s_2\}$, typically the side that loses under y . We use the full transcript D rather than isolated arguments from D , because debate quality depends on cross-turn context: claims, rebuttals, concessions, and unresolved clashes may only be interpretable through their interaction across speeches. Thus, f treated as an intervention on one side’s debate performance: to be useful, f should identify why s lost a specific comparative clash and suggest a revision that would make s more competitive.

3.2 Constructing Feedback Principles

Given a set of training debates $\mathbf{D} = \{D_1, \dots, D_n\}$, REFEEDGRAPH presents each debate’s motion, participating sides, and full transcript to five framework-specific LLM judges:

- NATIONALSPEECHANDDEBATEASSOCIATION (NSDA), which emphasizes contentions, evidence, clash, refutation, organization, and weighing (Melamed, 2026);
- UNITEDASIANDEBATINGCHAMPIONSHIP (UADC), which follows Asian Parliamentary norms and evaluates burdens, definitions, responsiveness, persuasiveness, and comparative weighing (Bengzon et al., 2025);
- TOULMIN, which evaluates claims, grounds, warrants, backing, qualifiers, and rebuttals (Toulmin, 1958);
- ROGERIAN, which evaluates fair representation, common ground, calibrated concessions, and constructive resolution (Teich, 1987); and

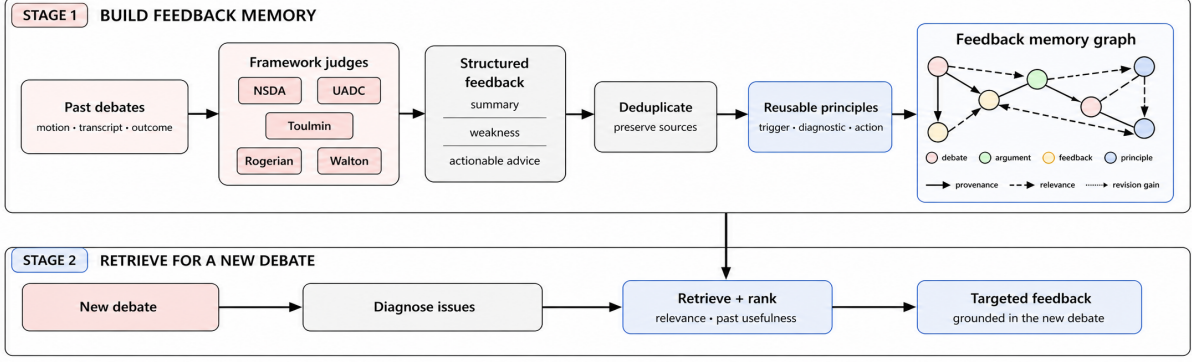


Figure 3: REFEEDGRAPH builds a feedback graph from prior critiques and retrieves relevant principles to generate targeted revision feedback.

- **WALTON**, which evaluates argument schemes and their critical questions (Walton et al., 2008).

Each judge receives D , selects a winner side s^+ and loser side s^- under its framework, writes a decision y , and gives grounded feedback f to each side.

Each raw judge feedback f is then structured through deterministic LLM prompting. REFEEDGRAPH splits f into sentence-level units and classifies each sentence into one of three categories, preserving the original judgment rather than synthesizing new feedback:

- **SUMMARY**;
- **WEAKNESS**; or
- **ACTIONABLEFEEDBACK**

It then deduplicates feedback within each category, merging only comments that express the same underlying failure or repair.

Finally, REFEEDGRAPH abstracts only **WEAKNESS** and **ACTIONABLEFEEDBACK** clusters into reusable principles; **SUMMARY** clusters are excluded because they describe debate content but do not encode transferable repairs. Each principle contains source feedback, source context, argument anchor, issue label, failure pattern, applicability conditions, diagnostic question, action template, and repair steps.

3.3 Constructing the Feedback Graph

REFEEDGRAPH stores these objects in a graph $G = (V, E)$ that preserves the path from debate evidence to reusable feedback. The graph contains five types of nodes:

- **SPEECHINSTANCE** nodes $V_s \subset V$ for speeches or sides receiving feedback;
- **SPECIFICFEEDBACK** nodes $V_f \subset V$ for sentence-level weaknesses and actionable feedback;
- **FEEDBACKCLUSTER** nodes $V_c \subset V$ for deduplicated feedback groups;
- **PRINCIPLE** nodes $V_p \subset V$ for abstract repair principles; and
- **ISSUELABEL** $V_i \subset V$ nodes for recurring failure categories such as missing warrant, insufficient weighing, dropped clash, weak evidence use, or unfair characterization.

Edges E encode the transformation from concrete feedback to reusable principles:

- **RECEIVEDFEEDBACK** edges $E_f \subset E$ links speeches to specific feedback;
- **PARTOFCLUSTER** edges $E_c \subset E$ links specific feedback to deduplicated clusters;
- **ABSTRACTEDTO** edges $E_p \subset E$ links clusters to principles;
- **INDICATESISSUE** links $E_i \subset E$ clusters to issue labels;
- **ADDRESSESISSUE** edges $E_a \subset E$ links principles to the issues they address; and
- **RELATEDTO** edges $E_r \subset E$ links principles sharing an issue label.

Per-debate graphs $G_1, \dots, G_n$ are merged into one feedback memory graph G_D while preserving debate identifiers, judge provenance, cluster memberships, and argument anchors.

3.4 Retrieving and Adapting Feedback

Given a debate transcript D at test time, REFEEDGRAPH first diagnoses the target weaker side s^- by prompting an LLM with D and asking for concrete issues with debate argument anchors. These issues become retrieval queries into the feedback memory graph G_D . Retrieval searches over principle and issue-label content using either of two retrieval methods:

- *Embedding retrieval* compares diagnosed issues to graph-node text, including principle text, issue labels, applicability conditions, and source contexts.
- *Lexical retrieval* uses token overlap, stopword removal, F-score-style overlap, and a phrase bonus. We also experiment with a REFEEDGRAPH+*specific* variant that additionally retrieves supporting examples from the original judge feedback linked through the graph.

The feedback generator adapts retrieved principles to the current debate rather than copying them directly. It is instructed to write only to s^- , preserve the side’s strongest existing strategy, retain useful material, avoid imposing new burdens, and rank recommendations by likely effect on the judge decision. Each feedback item f must specify the problem, why it matters, what to preserve or strengthen, and a concrete rewrite move.

4 Experimental Setup

Data We evaluate REFEEDGRAPH on Panel-Bench DebateArt and British Parliamentary (BP) (Liang et al., 2024), and Intelligence Squared (IQ2) debates (Zhang et al., 2016). DebateArt contains online one-versus-one debates with motion files, speech files, and gold outcomes. BP is restricted to Opening Government versus Opening Opposition to produce a two-sided setting. Debate where Closing Government or Closing Opposition won are excluded. IQ2 contains public Intelligence Squared debates represented as structured transcripts. For DebateArt we use 20 debates for training and 74 for testing. BP debates yield 7 training and 7 test debates. For IQ2, we use 20 debates for training and 70 for testing.

Baselines and Ablations We compare against direct and non-graph feedback baselines:

- LLM-PLAIN, which directly asks an LLM for feedback;

- LLM-STRUCTURED, which uses a structured feedback prompt without graph memory;
- PROF-FACTOR INSPIRED JUDGE, which uses persuasiveness, eloquence, evidence, correctness, relevance, and specificity. The judge is not fine-tuned according to (Nair et al., 2024), but is prompted using their method and factors of judgment.;
- MULTI-JUDGE FEEDBACK, which synthesizes two independent LLM feedback reports; and saved PROF feedback when available.

We also evaluate ablations of REFEEDGRAPH that isolate the effects of different parts of the approach:

- REFEEDGRAPH-NO-SPECIFIC-EXAMPLES. Effect of source examples;
- REFEEDGRAPH-NO-GRAPH. Effect of graph memory;
- REFEEDGRAPH-LEXICAL and REFEEDGRAPH-EMBEDDING. Effect of retrieval method;
- REFEEDGRAPH-RANDOM. Effect of relevance; and
- REFEEDGRAPH-RAW-PRINCIPLES Effect of debate-specific adaptation.

Models and Implementation All model calls use temperature 0. We separate the following model roles:

- **Feedback generation.** The main experiments use gpt-4o-mini; robustness experiments additionally use openai/gpt-oss-20b and Llama 3.1 8B. Within each run, all methods receive the same debate and target side.
- **Speech rewriting.** All conditions use the same fixed gpt-4o-mini rewriter, so only the feedback method varies.
- **Revision evaluation.** Tables 1–7 use gpt-4o-mini to score the original and rewritten speeches. Appendix Table 8 additionally reports GPT-4.1 results on all 74 DebateArt cases and partial Claude results on 25 cases.
- **Feedback-quality evaluation.** GPT-4.1-mini compares anonymized feedback for specificity, actionability, and helpfulness.

We use text-embedding-3-small for embedding retrieval. Structured stages request JSON outputs and retry or repair malformed responses.

Rewriting-Based Evaluation Our primary evaluation measures whether a feedback item f improves the weaker side s^- after revision. For each held-out debate and feedback condition, a rewriter receives the motion m , full transcript D , target side $s \in \{s_1, s_2\}$, original target speech, and f . It rewrites only the target speech and is instructed to preserve the original position, organization, evidence, examples, citations, and voice unless the feedback motivates a change.

We score the original weaker-side speech, the original stronger-side speech, and each revised weaker-side speech in full debate context. Similar to Nair et al. (2024), the evaluator assigns 0-5 scores for persuasiveness, eloquence, evidence, correctness, relevance, and specificity, which are averaged into an overall score. We report the mean score of the original weaker-side speech, the mean score of the rewritten speech, and the mean improvement after revision:

$$\Delta_{\text{weak}} := \text{score}(\hat{s}_{\text{weak}}) - \text{score}(s_{\text{weak}}).$$

We also report the *improved-case rate*, defined as the percentage of debates for which

$$\text{score}(\hat{s}_{\text{weak}}) > \text{score}(s_{\text{weak}}).$$

To measure whether a revision closes or reverses the original score gap, we report two round-level outcomes. A *strict winner flip* occurs when

$$\text{score}(\hat{s}_{\text{weak}}) > \text{score}(s_{\text{strong}}),$$

whereas a *tie-inclusive takeover* occurs when

$$\text{score}(\hat{s}_{\text{weak}}) \geq \text{score}(s_{\text{strong}}).$$

Appendix D additionally reports the evaluator’s alignment with the dataset-recorded outcomes and repeats the winner-flip analysis on the subset of debates for which the evaluator initially scores the dataset-recorded winner strictly above the loser.

Feedback Quality Evaluation We additionally evaluate generated feedback with an LLM-as-judge using OpenAI GPT-4.1-mini. Given the full transcript D , target weaker side s^- , and feedback f , the judge rates the following three criteria.

- *Specificity*. Measures grounding in concrete debate material;
- *Helpfulness*. Measures whether the advice targets likely sources of improvement; and

Method	Avg. original	Avg. rewritten	Avg. Δ	Improved cases
<i>PanelBench (DebateArt)</i>				
REFEEDGRAPH+specific	3.216	3.707	0.491	82.4%
REFEEDGRAPH	3.216	3.703	0.486	85.1%
LLM-PLAIN	3.216	3.532	0.315	73.0%
LLM-STRUCTURED	3.216	3.547	0.331	71.6%
PROF-FACTOR JUDGE	3.216	3.658	0.441	86.5%
MULTI-JUDGE FEEDBACK	3.216	3.554	0.338	66.2%
<i>IQ2</i>				
REFEEDGRAPH+specific	3.914	4.148	0.233	61.4%
REFEEDGRAPH	3.914	4.169	0.255	58.6%
LLM-PLAIN	3.914	4.129	0.214	55.7%
LLM-STRUCTURED	3.914	4.121	0.207	48.6%
PROF-FACTOR JUDGE	3.914	4.136	0.221	50.0%
MULTI-JUDGE FEEDBACK	3.914	4.138	0.224	57.1%
<i>PanelBench (BP OG/OO)</i>				
REFEEDGRAPH+specific	3.619	3.714	0.095	57.1%
REFEEDGRAPH	3.619	3.857	0.238	57.1%
LLM-PLAIN	3.619	3.738	0.119	42.9%
LLM-STRUCTURED	3.619	3.810	0.190	57.1%
PROF-FACTOR JUDGE	3.619	3.762	0.143	57.1%
MULTI-JUDGE FEEDBACK	3.619	3.810	0.190	42.9%

Table 1: **Feedback method performance across datasets.** Avg. *original* is the mean score of the original losing-side speech before revision. Avg. *rewritten* is the mean score after applying each feedback method and rewriting the speech. Avg. Δ is the mean rewritten-minus-original score change. *Improved cases* reports the percentage of test debates for which the rewritten speech scored higher than the original.

- *Actionability*. Measures whether the feedback gives implementable revision moves.

We report these metrics separately for DebateArt, BP, and IQ2, as well as averaged across datasets.

5 Results

Feedback Method Performance Table 1 shows that our feedback method is effective across datasets, suggesting that it is not tied to a specific debate format. Our two REFEEDGRAPH variants obtain the largest average gains on PanelBench (DebateArt) and IQ2, and REFEEDGRAPH achieves the strongest result on PanelBench (BP). The improvement is especially large on DebateArt, where REFEEDGRAPH+*specific* reaches the highest rewritten score (3.707) and gain ($\Delta = 0.491$), while REFEEDGRAPH improves 85.1% of cases. On IQ2, where the original scores are already higher, REFEEDGRAPH still gives the best rewritten score (4.169) and largest gain ($\Delta = 0.255$). Even when PROF-based feedback optimizes for PROF evaluation criteria, our method outperforms it. Overall, these results indicate that our approach provides consistent, dataset-agnostic improvements over LLM-only and judge-based feedback baselines.

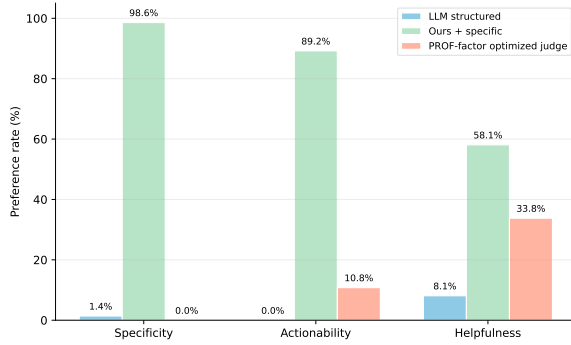


Figure 4: **Feedback preference rates.** Meta-judged preferences for specificity, actionability, and helpfulness across feedback methods on DebateArt.

Feedback condition	Mean words	Avg. Δ	Improved	Winner takeover
ReFEEDGRAPH+specific	784.2	+0.545	92.3%	69.2%
Normal ReFEEDGRAPH	777.3	+0.513	92.3%	65.4%
Length-controlled ReFEEDGRAPH	521.8	+0.487	88.5%	57.7%
PROF-FACTOR JUDGE	529.0	+0.455	92.3%	61.5%
MULTI-JUDGE FEEDBACK	504.4	+0.333	80.8%	61.5%
LLM-STRUCTURED	357.5	+0.333	84.6%	53.8%
LLM-PLAIN	443.0	+0.327	84.6%	53.8%

Table 2: **Length-controlled feedback results on the matched 26-case DebateArt subset.** *Mean words* is the average feedback length. *Avg. Δ* is the mean rewritten-minus-original score change. *Improved* reports rewrites with a positive score gain. *Winner takeover* is tie-inclusive ($\geq$).

Feedback Preference Analysis Figure 4 compares feedback preferences along specificity, actionability, and helpfulness. Our variant REFEEDGRAPH+*specific* is strongly preferred on specificity (98.6%) and actionability (89.2%), showing that the generated feedback gives more concrete and usable revision guidance than the baselines. The gap is smaller for helpfulness, but our method is still preferred most often (58.1%), outperforming both LLM-STRUCTURED feedback and the PROF-FACTOR JUDGE. These results suggest that our method’s main advantage lies in producing feedback that is detailed enough to guide targeted revisions, rather than only offering high-level judgments.

Ablation Analysis. Figure 5 shows that the method remains effective across ablations, but performance drops as feedback becomes less structured. The strongest result comes from using graph principles without source examples, achieving the highest score gain ($\Delta = 0.493$) and improved-case rate (88%). Variants that use lexical or embedding-based graph retrieval remain close to

the full method, suggesting that the approach is robust to the retrieval strategy. In contrast, using only raw retrieved principles or LLM-framework judges leads to lower gains, indicating that structured graph-based feedback contributes substantially to the effectiveness of the revisions.

Feedback Length and Revision Outcomes. Table 2 shows that longer feedback is generally associated with stronger revisions, but length alone does not explain performance. Our two full-length variants achieve the largest average gains and highest winner-takeover rates, with REFEEDGRAPH+*specific* performing best overall. The length-controlled variant remains competitive with similarly sized baselines, suggesting that the structure and specificity of the feedback contribute beyond length alone. Overall, our feedback is particularly effective at producing revisions strong enough to change the debate outcome.

Robustness across Feedback Models Table 3 shows that our feedback structure remains competitive across different feedback-generation models. With GPT-4o-mini, REFEEDGRAPH+*specific* achieves the largest average gain among the compared methods and is overwhelmingly preferred for specificity and actionability, while also receiving the highest helpfulness rate. With OSS-20B, REFEEDGRAPH+*specific* retains a substantially larger average gain and is preferred for specificity and actionability, while LLM-STRUCTURED is preferred for helpfulness. With Llama 3.1 8B, all methods produce smaller score gains, but REFEEDGRAPH+*specific* is still strongly preferred on all three feedback-quality dimensions. Overall, the results suggest that our method reliably improves the structure and usability of feedback across model families, even when score gains vary with the underlying generator.

Winner-Flip Rate Table 4 shows that our methods consistently achieve the strongest or tied-strongest strict-flip and takeover performance across datasets, indicating that their gains often close or reverse the evaluator-assigned gap between the original winner and loser. This pattern remains in the stricter analysis in Appendix D Table 10, which includes only debates where the evaluator initially ranks the dataset-recorded (gold) winner above the loser.

Effect of Graph Size Table 5 shows that performance is stable across graph sizes. Both variants

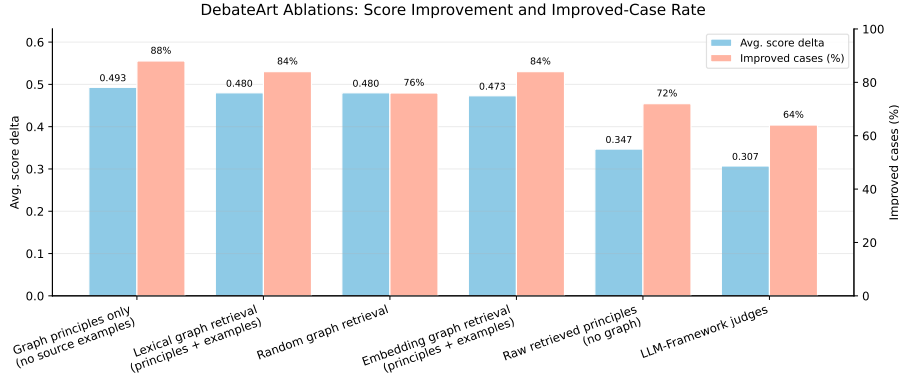


Figure 5: **DebateArt ablations.** Avg. score Δ and improved cases % for six feedback graph variants on DebateArt.

Model	Method	Avg. Δ	Improved	Strict $>$	Takeover $\geq$	Most specific	Most actionable	Most helpful
<i>GPT-4o-mini feedback run</i>								
GPT-4o-mini	REFEEDGRAPH+specific	0.491	82.4%	52.7%	59.5%	95.9%	93.2%	58.1%
GPT-4o-mini	LLM-STRUCTURED	0.331	71.6%	43.2%	51.4%	1.4%	0.0%	9.5%
GPT-4o-mini	PROF-FACTOR JUDGE	0.441	86.5%	48.6%	55.4%	2.7%	6.8%	32.4%
<i>GPT-OSS-20B feedback run</i>								
GPT-OSS-20B	REFEEDGRAPH+specific	0.667	90.5%	52.7%	60.8%	55.9%	73.5%	5.9%
GPT-OSS-20B	LLM-STRUCTURED	0.444	74.3%	45.9%	55.4%	20.6%	11.8%	58.8%
GPT-OSS-20B	PROF-FACTOR JUDGE	0.475	77.0%	47.3%	58.1%	23.5%	14.7%	35.3%
<i>Llama 3.1 8B feedback run</i>								
Llama 3.1 8B	REFEEDGRAPH+specific	0.423	77.0%	41.9%	52.7%	95.0%	95.0%	87.5%
Llama 3.1 8B	LLM-STRUCTURED	0.446	81.1%	43.2%	54.1%	2.5%	2.5%	10.0%
Llama 3.1 8B	PROF-FACTOR JUDGE	0.423	77.0%	43.2%	56.8%	2.5%	2.5%	2.5%

Table 3: **Performance and feedback-quality judgments across model runs.** Avg. Δ is the mean rewritten-minus-original score change. *Improved* reports the percentage of cases where the rewritten speech scored higher than the original. *Strict $>$* reports rewrites that exceeded the original winning-side speech; *Takeover $\geq$* also includes ties. *Most specific*, *most actionable*, and *most helpful* report GPT-4.1-mini meta-judgment win rates (separate run from Fig 4) among REFEEDGRAPH+specific, LLM-STRUCTURED, and PROF-FACTOR JUDGE feedback.

improve the original speeches for at least 70% of cases in every setting, and the average gains remain in a narrow range. Larger graphs do not consistently improve performance, suggesting that a compact graph can already capture useful feedback principles for DebateArt. This shows the utility of our method which makes it cost efficient.

6 Discussion

We designed REFEEDGRAPH with the following principles in mind: (1) explainability, (2) data efficiency, and (3) reusability. Since REFEEDGRAPH generates feedback from the graph, every output can be audited by comparing the output to the snapshot of the graph at the point of time at which the output was generated. This can enable the construction of automatic explainer systems that can generate clear, natural language explanations of why certain feedback was generated by performing com-

putations over the graph and the generated feedback. We leave the construction of such a system to future work. REFEEDGRAPH uses in-context learning, which enables more efficient use of the training data over traditional approaches like model training. This is especially relevant in the field of argumentation, where the number of large, high quality datasets that can support model training approaches is small. A drawback of the in-context learning approach is the prompting overhead required to construct the initial feedback memory graph. However, the constructed graph can be used not just to enable faster feedback generation over subsequent runs; but can also be used by other systems to improve the quality of their outputs. Thus, the tradeoff between the overhead and the reusability of our approach makes the initial investment worthwhile.

Method	Strict >	Takeover $\geq$
<i>DebateArt</i>		
REFEEDGRAPH+specific	52.7%	59.5%
REFEEDGRAPH	48.6%	58.1%
LLM-PLAIN	44.6%	51.4%
LLM-STRUCTURED	43.2%	51.4%
PROF-FACTOR JUDGE	48.6%	55.4%
MULTI-JUDGE FEEDBACK	45.9%	55.4%
<i>IQ2</i>		
REFEEDGRAPH+specific	58.6%	68.6%
REFEEDGRAPH	57.1%	71.4%
LLM-PLAIN	58.6%	67.1%
LLM-STRUCTURED	52.9%	67.1%
PROF-FACTOR JUDGE	55.7%	67.1%
MULTI-JUDGE FEEDBACK	52.9%	62.9%
<i>BP OG/OO</i>		
REFEEDGRAPH+specific	42.9%	85.7%
REFEEDGRAPH	57.1%	85.7%
LLM-PLAIN	42.9%	71.4%
LLM-STRUCTURED	42.9%	85.7%
PROF-FACTOR JUDGE	42.9%	85.7%
MULTI-JUDGE FEEDBACK	42.9%	71.4%

Table 4: **Strict winner-flip and tie-inclusive takeover rates after feedback.** Strict winner flip requires the rewritten losing-side speech to exceed the original winning-side score ($>$). Tie-inclusive takeover also counts equality ($\geq$).

Setting		Rewrite outcome		
Size	Method	Score	Δ	Improved
5	REFEEDGRAPH+specific	3.692	0.542	80.0%
	REFEEDGRAPH	3.725	0.575	85.0%
9	REFEEDGRAPH+specific	3.683	0.508	70.0%
	REFEEDGRAPH	3.708	0.533	75.0%
20	REFEEDGRAPH+specific	3.633	0.583	80.0%
	REFEEDGRAPH	3.558	0.508	85.0%

Table 5: **Effect of graph size on DebateArt.** All graph sizes are evaluated on the same 20 unseen DebateArt test cases. *Score* is the mean rewritten score, Δ is rewritten-minus-original score change, and *improved* is the percentage of cases where the rewritten speech scored higher than the original.

Case Study: “The Bible has proven the existence of God” This case supports our broader claim that LLMs can often recognize argumentative issues, but strong feedback requires a memory system that turns those issues into effective repairs. In this debate, all methods noticed the same general problems: the debater was repetitive, relied on circular reasoning, lacked objective evidence, and failed to answer Con’s burden-of-proof objection. However, recognition alone was not enough. The LLM-PLAIN, LLM-STRUCTURED, and PROF-FACTOR JUDGE baselines each improved the average score by only +0.33, while multi-judge feedback improved it by +0.20. In contrast, our graph-retrieved feedback plus supporting examples improved the average score by +2.67.

This is the central value of the feedback memory graph. LLMs are already reasonably good at saying what is wrong with an argument. What they lack, especially in zero-shot contexts, is a stable source of accumulated knowledge: which kinds of feedback have helped in similar cases, what level of specificity is needed, and how to preserve the debater’s strategy while fixing the failure. REFEEDGRAPH supplies that missing layer by retrieving principles learned from prior debates and adapting them to the current one.

The baselines gave plausible advice such as “add evidence,” “avoid circular reasoning,” and “address the opponent,” but these recommendations were underspecified. Our feedback converted the diagnosis into concrete steps that could be followed: define what “proof” means, connect specific biblical passages to that standard, break the circular claim that believers have proof because the Bible says so, add independently assessable support, and directly answer Con’s objection that the Bible is not a reliable source. This resulted in a feedback that scored highly on the actionable metric and was thus able to outperform the baseline in terms of score improvement.

7 Conclusion

We introduce REFEEDGRAPH, a graph-based method that turns debate feedback into reusable principles. Instead of generating feedback from scratch, it retrieves relevant failure–repair patterns and adapts them into targeted feedback for the weaker side. Across three different datasets, REFEEDGRAPH improves speeches by producing feedback that is more specific and actionable. These results suggest that debate feedback is stronger when models reuse what has helped before, rather than treating each debate as isolated.

8 Limitations

Our study focuses on debate feedback for two-sided debate settings, where the system receives a full transcript, identifies the weaker side, and generates revision-oriented feedback for that side. This setup is appropriate for evaluating whether feedback can help a losing side become more competitive, but it does not cover all forms of debate or argumentation. For example, multi-party debates, informal online discussions, classroom peer review, live spoken debate, and debates without clear winner–loser labels may require different feedback strategies.

We partially address this by evaluating across DebateArt, Intelligence Squared, and British Parliamentary debates, but broader validation is needed before treating the method as general to all argumentative settings.

Another limitation is that our evaluation relies on rewrite-based improvement rather than direct human debater outcomes. This follows prior work that evaluates feedback through its effect on revision, but the revised speeches are produced by a model rather than by human debaters. As a result, the measured gains reflect whether the feedback is useful to a rewriting model under controlled conditions, not necessarily whether a student, debater, or coach would use the feedback in the same way. To reduce this concern, we evaluate the rewritten speech in full debate context and also report feedback-quality dimensions such as specificity, helpfulness, and actionability.

The feedback graph is also built from a limited number of training debates. Although our graph-size analysis suggests that compact graphs can already provide useful reusable principles, some debates may require principles that are missing or only weakly represented in the graph. In such cases, retrieval can surface vaguely relevant principles that do not fully address the central weakness of the speech. We mitigate this by instructing the generator to treat retrieved principles as candidate memories rather than obligations, and by preserving debate-specific diagnosis before retrieval.

Finally, some stages of the pipeline depend on LLM judgments, including framework-specific judging, sentence classification, principle abstraction, feedback generation, rewriting, and evaluation. This may introduce model-specific bias or reward feedback that is especially easy for LLM rewriters and evaluators to use. We partially address this by using multiple judging frameworks, structured intermediate representations, deterministic decoding, baseline comparisons, ablations, and model-robustness experiments. Still, future work should include more human evaluation and real debater studies.

Potential Risks This work could be misused to produce persuasive debate coaching that improves rhetorical effectiveness without improving truthfulness, fairness, or epistemic quality. To reduce this risk, our prompts emphasize concrete clash engagement, fair treatment of the opponent, preservation of the debater’s original position, and avoidance

of invented evidence. Another risk is that graph-grounded feedback may appear more authoritative because it is retrieved from prior principles. We therefore frame the graph as a reusable feedback memory, not as a source of definitive debate truth. Retrieved principles should be treated as guidance that must be adapted to the current debate, checked against the transcript, and rejected when irrelevant.

Reproducibility We report the datasets, train/test construction, baselines, ablations, feedback-generation models, decoding settings, retrieval methods, rewriting protocol, and evaluation metrics used in the experiments. The method uses deterministic decoding with temperature 0.0, structured JSON outputs where possible, and retry/repair fallbacks for malformed generations. We also provide the main prompts for framework-specific judging, sentence-level classification, deduplication, principle abstraction, diagnosis, graph-conditioned feedback generation, rewriting, and evaluation. Some exact costs may vary by provider and model pricing, but we report approximate call counts and cost relative to direct prompting to make the scale of the pipeline clear. The models used for the experiments in this paper were accessed through the Groq platform ([Groq](#)).

9 Ethical Considerations

Our work aims to improve the quality of debate feedback by making it more specific, actionable, and grounded in prior critique patterns. The datasets used are existing debate datasets, and our analysis focuses on debate texts and system-generated feedback rather than private user data. However, debate feedback systems can influence how people argue, persuade, and frame disagreement. For this reason, such systems should not encourage fabricated evidence, manipulative rhetoric, personal attacks, or unfair characterization of opponents. In our pipeline, the feedback generator is instructed to preserve the debater’s position while improving clash engagement, evidence use, warranting, burden framing, and weighing. For educational or competitive settings, the system should be used as an assistive tool for revision and coaching, not as a replacement for human judgment, debate instruction, or tournament adjudication.

AI Assistance in Writing AI writing tools were used to assist with editing, wording refinement, and organization of the manuscript. All technical

claims, experimental results, analyses, and final writing decisions were reviewed and controlled by the authors. AI assistance was not used as a substitute for conducting experiments, validating results, or determining the paper’s scientific conclusions.

References

- Shengnan An, Zexiong Ma, Zeqi Lin, Nanning Zheng, Jian-Guang Lou, and Weizhu Chen. 2023. [Learning from mistakes makes LLM better reasoner](#). *Preprint*, arXiv:2310.20689.
- Jaed Bengzon, Vansh Chadha, Ahmad Tousif Jami, Lim Xing Han Wayne, Muhammad Batara Mulya, and Anas Bin Ahmad Zaharim. 2025. [Guangzhou UADC Judging Manual](#).
- Eric Chamoun, Michael Schlichtkrull, and Andreas Vlachos. 2024. [Automated focused feedback generation for scientific writing assistance](#). In *Findings of the Association for Computational Linguistics: ACL 2024*.
- Seongyeub Chu, Jongwoo Kim, and Munyong Yi. 2026. Feedeval: Pedagogically aligned evaluation of llm-generated essay feedback. *arXiv preprint arXiv:2601.04574*.
- Mike D’Arcy, Tom Hope, Larry Birnbaum, and Doug Downey. 2024. MARG: Multi-Agent Review Generation for Scientific Papers. *arXiv preprint arXiv:2401.04259*.
- Jeffrey P. Mehlretter Drury. 2019. *Argumentation in Everyday Life*. SAGE Publications, Thousand Oaks, CA.
- Thomas Gordon, Henry Prakken, and Douglas Walton. 2007. [The carneades model of argument and burden of proof](#). *Artificial Intelligence*, 171(10):875–896.
- Groq. Groq. <https://groq.com/>. Accessed: 2026-08-31.
- Amy Gutmann and Dennis Thompson. 2004. [Why Deliberative Democracy?](#) Princeton University Press, Princeton, NJ.
- 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.
- Jingcong Liang, Rong Ye, Meng Han, Ruofei Lai, Xinyu Zhang, Xuan-Jing Huang, and Zhongyu Wei. 2024. Debatrix: Multi-dimensional debate judge with iterative chronological analysis based on llm. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 14575–14595.
- Aman Madaan and 1 others. 2023. [Self-refine: Iterative refinement with self-feedback](#). *Preprint*, arXiv:2303.17651.
- Jordan Melamed. 2026. [How To Preside: 101](#).
- Haile Misgna, Byung-Won On, Ingyu Lee, and Gyu Sang Choi. 2025. [A survey on deep learning-based automated essay scoring and feedback generation](#). *Artificial Intelligence Review*, 58(36).
- Inderjeet Nair, Jiaye Tan, Xiaotian Su, Anne Gere, Xu Wang, and Lu Wang. 2024. [Closing the loop: Learning to generate writing feedback via language model simulated student revisions](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 16636–16657, Miami, Florida, USA. Association for Computational Linguistics.
- Keyang Qian, Kaixun Yang, Wei Dai, Flora Jin, Yixin Cheng, Rui Guan, Sadia Nawaz, Zachari Swiecki, Guanliang Chen, Lixiang Yan, and 1 others. 2025. Scalefeedback: a large-scale dataset of synthetic computer science assignments for llm-generated educational feedback research. *arXiv preprint arXiv:2508.05953*.
- Hannah Rashkin, Elizabeth Clark, Fantine Huot, and Mirella Lapata. 2025. [Help me write a story: Evaluating LLMs’ ability to generate writing feedback](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 25827–25847, Vienna, Austria. Association for Computational Linguistics.
- Noah Shinn, Federico Cassano, Edward Berman, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. [Reflexion: Language agents with verbal reinforcement learning](#). *Preprint*, arXiv:2303.11366.
- Maja Stahl, Leon Biermann, Andreas Nehring, and Henning Wachsmuth. 2024. [Exploring LLM prompting strategies for joint essay scoring and feedback generation](#). In *Proceedings of the 19th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2024)*, pages 283–298, Mexico City, Mexico. Association for Computational Linguistics.
- Laurie Taylor. 2013. [The language of argumentation: Using debate to spark interest and learning in science](#). *The Science Teacher*, 80(5):44–49.
- Nathaniel Teich. 1987. Rogerian problem-solving and the rhetoric of argumentation. *Journal of advanced composition*, pages 52–61.
- Nitya Thakkar, Mert Yuksekgonul, Jake Silberg, Animesh Garg, Nanyun Peng, Fei Sha, Rose Yu, Carl Vondrick, and James Zou. 2025. [Can llm feedback enhance review quality? a randomized study of 20k reviews at iclr 2025](#). *Preprint*, arXiv:2504.09737.
- Stephen E. Toulmin. 1958. *The Uses of Argument*. Cambridge University Press.

Douglas Walton, Christopher Reed, and Fabrizio Macagno. 2008. *Argumentation Schemes*. Cambridge University Press.

Thiemo Wambsganss, Sebastian Guggisberg, and Matthias Söllner. 2021. *Arguebot: A conversational agent for adaptive argumentation feedback*. In *Innovation Through Information Systems*, pages 267–282. Springer.

Danqing Wang and Lei Li. 2023. *Learning from mistakes via cooperative study assistant for large language models*. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*.

Xinyu Wang, Yohan Lee, and Juneyoung Park. 2022. *Automated evaluation for student argumentative writing: A survey*. Preprint, arXiv:2205.04083.

Justine Zhang, Ravi Kumar, Sujith Ravi, and Cristian Danescu-Niculescu-Mizil. 2016. Conversational flow in oxford-style debates. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 136–141.

Andrew Zhao, Daniel Huang, Quentin Xu, Matthieu Lin, Yong-Jin Liu, and Gao Huang. 2024. *ExpeL: LLM agents are experiential learners*. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

A Sample Graph

Figure 6 illustrates how the feedback graph connects low-level critique patterns to higher-level revision guidance. The graph groups related feedback failures into clusters, abstracts them into reusable principles, and links these principles to concrete issues in the target debate. This structure supports targeted revisions: for example, the highlighted debate is connected to principles about giving reasons and evidence, using the strongest framework, and weighing competing impacts, which then lead to actionable rewrite suggestions. The example shows how the graph can move beyond isolated comments by tracing a path from observed weaknesses to general feedback principles and then to concrete revision operations.

B Approximate Computational Cost

REFEEDGRAPH is more expensive than direct prompting. A direct LLM baseline only requires one feedback-generation call, while our full pipeline uses roughly 10–15 calls per debate when including diagnosis, retrieved-principle conditioning, feedback generation, rewriting, and evaluation. The graph construction stage also adds an

upfront cost because each training debate is processed through multiple framework-specific judges, sentence classification, deduplication, and principle abstraction. However, this cost buys reusability: the resulting graph can be reused across future debates, and our graph-size results suggest that even compact graphs can already produce stable improvements. In our experiments, the full set of runs cost approximately \$1 per debate, which we consider acceptable for high-quality revision feedback but higher than lightweight direct prompting baselines.

C Additional Results

Robustness across feedback models. Table 7 presents the complete revision and feedback-quality results across the three feedback-model runs. The graph-based methods are strongest overall with GPT-4o-mini and GPT-OSS-20B, achieving the largest or near-largest revision gains, improvement rates, and winner-takeover rates. Performance differences are smaller with Llama 3.1 8B, where no method dominates every outcome. Nevertheless, adding retrieved specific examples consistently produces feedback judged to be substantially more specific and actionable, although the preferred method for overall helpfulness varies across models. The meta-judgment results cover the available subsets for each run: 74 GPT-4o-mini, 34 GPT-OSS-20B, and 40 Llama 3.1 8B cases.

Robustness across evaluators. Table 8 evaluates the revisions using GPT-4.1 and Claude in place of the primary evaluator. The relative ranking of methods varies across evaluators: the PROF-factor baseline obtains the largest average gain and improvement rate under GPT-4.1, while different methods lead individual metrics on the partial Claude subset. However, REFEEDGRAPH+*specific* achieves the highest takeover rate under both evaluators, providing additional evidence that its feedback can produce revisions that reach the original winning-side score. Because the Claude evaluation covers only 25 cases, its results should be treated as preliminary rather than directly compared with the complete GPT-4.1 evaluation.

D Evaluator Alignment and Conditional Winner Flips

Alignment with recorded outcomes. The dataset-recorded winner and the evaluator’s preferred side need not coincide because DebateArt

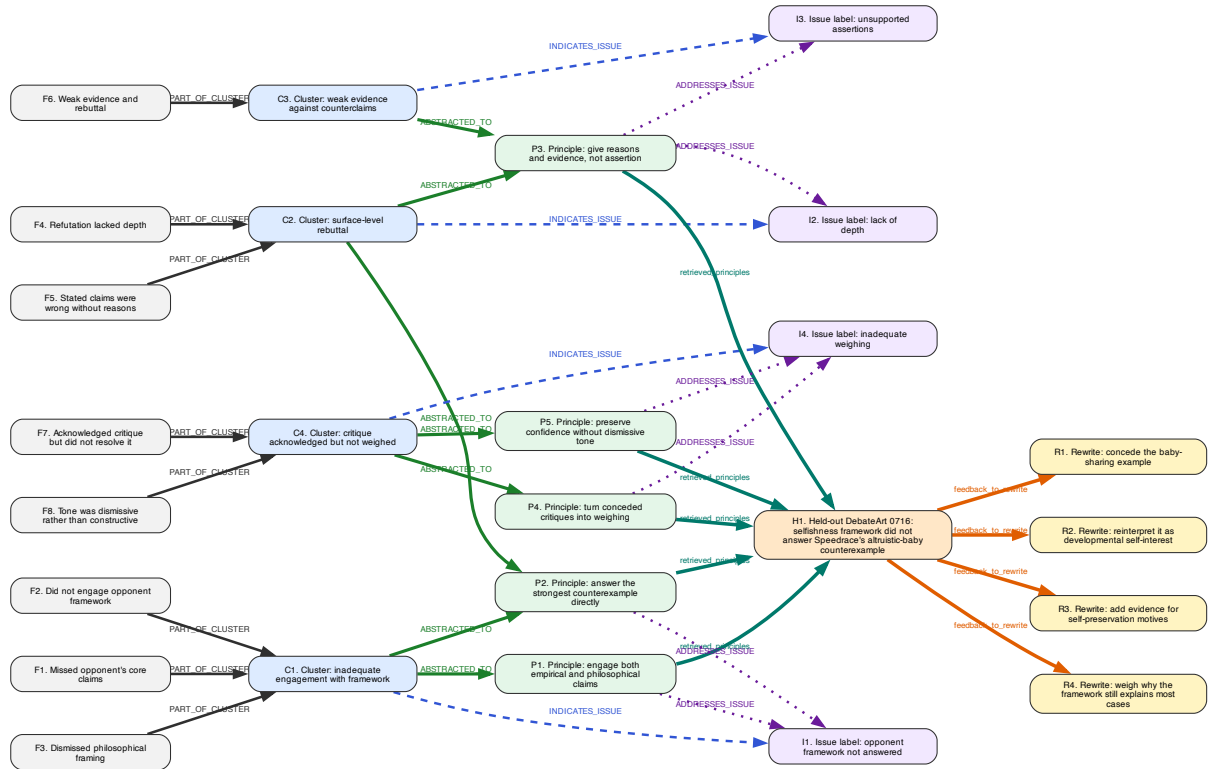


Figure 6: Graph Example

Method	Graph construction	Test-time calls / debate	Relative cost
LLM plain	—	1	1.0×
LLM structured	—	1	1.0–1.2×
PROF-factor judge	—	1	1.0–1.3×
Multi-judge feedback	—	2–3	2–3×
REFEEDGRAPH	10–15 calls / training debate	10–15	~10–15×
REFEEDGRAPH + specific	10–15 calls / training debate	10–15	~12–18×

Table 6: Approximate computational cost relative to direct LLM feedback. REFEEDGRAPH requires substantially more calls because it performs diagnosis, graph retrieval, feedback generation, rewriting, and evaluation, in addition to the preprocessing calls used to construct the reusable feedback graph.

and IQ2 outcomes reflect platform or audience voting, whereas our evaluator applies a fixed six-dimensional rubric. Table 9 quantifies this difference. The evaluator assigns the recorded winner an equal or higher score in a majority of cases across all three datasets, although strict agreement is substantially lower for IQ2 and BP. We therefore interpret takeover as exceeding the evaluator-assigned score of the dataset-recorded winner, rather than as reproducing the original human voting process.

Conditional winner-flip analysis. To test whether our takeover results depend on disagreements between the evaluator and the recorded outcome, Table 10 restricts the analysis to debates where the evaluator initially assigns the recorded winner a strictly higher score than the loser. Under this more conservative evaluation, the

REFEEDGRAPH variants remain strongest or tied for strongest on DebateArt and IQ2. This shows that their advantage is not driven only by cases where the evaluator already preferred the recorded loser. The BP subset contains only three aligned debates, so its conditional results are descriptive and do not support reliable method comparisons.

E Additional Case Study

“Violent revolution is a just response to political oppression” The delta improvement of our method is minimal when the principles are only vaguely relevant and not a single principle precisely matches the improvement that needs to be made. In such cases, the retrieval sometimes targets principles that result in feedback that does not fully address the core weaknesses of the speech. For

Model	Method	Avg. Δ	Improved	Strict $>$	Takeover $\geq$	Most specific	Most actionable	Most helpful
<i>GPT-4o-mini feedback run</i>								
GPT-4o-mini	REFEEDGRAPH+specific	0.491	82.4%	52.7%	59.5%	95.9%	93.2%	58.1%
GPT-4o-mini	REFEEDGRAPH	0.486	85.1%	48.6%	58.1%	—	—	—
GPT-4o-mini	LLM-PLAIN	0.315	73.0%	44.6%	51.4%	—	—	—
GPT-4o-mini	LLM-STRUCTURED	0.331	71.6%	43.2%	51.4%	1.4%	0.0%	9.5%
GPT-4o-mini	PROF-FACTOR JUDGE	0.441	86.5%	48.6%	55.4%	2.7%	6.8%	32.4%
GPT-4o-mini	MULTI-JUDGE FEEDBACK	0.338	66.2%	45.9%	55.4%	—	—	—
<i>GPT-OSS-20B feedback run</i>								
GPT-OSS-20B	REFEEDGRAPH+specific	0.667	90.5%	52.7%	60.8%	55.9%	73.5%	5.9%
GPT-OSS-20B	REFEEDGRAPH	0.676	91.9%	56.8%	64.9%	—	—	—
GPT-OSS-20B	LLM-PLAIN	0.423	74.3%	45.9%	55.4%	—	—	—
GPT-OSS-20B	LLM-STRUCTURED	0.444	74.3%	45.9%	55.4%	20.6%	11.8%	58.8%
GPT-OSS-20B	PROF-FACTOR JUDGE	0.475	77.0%	47.3%	58.1%	23.5%	14.7%	35.3%
GPT-OSS-20B	MULTI-JUDGE FEEDBACK	0.477	77.0%	47.3%	59.5%	—	—	—
<i>Llama 3.1 8B feedback run</i>								
Llama 3.1 8B	REFEEDGRAPH+specific	0.423	77.0%	41.9%	52.7%	95.0%	95.0%	87.5%
Llama 3.1 8B	REFEEDGRAPH	0.430	79.7%	44.6%	55.4%	—	—	—
Llama 3.1 8B	LLM-PLAIN	0.439	77.0%	41.9%	55.4%	—	—	—
Llama 3.1 8B	LLM-STRUCTURED	0.446	81.1%	43.2%	54.1%	2.5%	2.5%	10.0%
Llama 3.1 8B	PROF-FACTOR JUDGE	0.423	77.0%	43.2%	56.8%	2.5%	2.5%	2.5%
Llama 3.1 8B	MULTI-JUDGE FEEDBACK	0.421	78.4%	41.9%	52.7%	—	—	—

Table 7: **Full performance and feedback-quality results across model runs.** Avg. Δ is the mean rewritten-minus-original score change. *Improved* reports the percentage of cases where the rewritten speech scored higher than the original. *Strict $>$* reports rewrites that exceeded the original winning-side speech; *Takeover $\geq$* also includes ties. *Most specific*, *most actionable*, and *most helpful* report GPT-4.1-mini meta-judgment win rates. Dashes indicate methods that were not included in the three-way meta-judgment comparison.

Method	GPT-4.1 ($n = 74$)			Claude ($n = 25$)		
	Avg. Δ	Improved	Takeover	Avg. Δ	Improved	Takeover
REFEEDGRAPH+specific	+0.295	62.2%	33.8%	+0.073	36.0%	40.0%
REFEEDGRAPH	+0.313	66.2%	27.0%	+0.060	40.0%	32.0%
LLM-PLAIN	+0.162	48.6%	27.0%	-0.020	32.0%	32.0%
LLM-STRUCTURED	+0.212	56.8%	28.4%	+0.033	32.0%	32.0%
PROF-FACTOR JUDGE	+0.356	73.0%	31.1%	+0.067	32.0%	36.0%
MULTI-JUDGE FEEDBACK	+0.252	56.8%	29.7%	+0.013	44.0%	36.0%

Table 8: **Revision outcomes under additional evaluators.** Avg. Δ is the mean rewritten-minus-original score change. *Improved* reports rewrites with a positive score gain. *Takeover* reports rewrites that matched or exceeded the original winning-side score ($\geq$). GPT-4.1 covers the full 74-case DebateArt test set; Claude results cover the completed 25-case subset and should be interpreted as a partial evaluation.

Dataset	n	Gold winner scored higher	Equal score	Gold winner higher or tied
DebateArt	74	48 (64.9%)	5 (6.8%)	53 (71.6%)
IQ2	70	34 (48.6%)	9 (12.9%)	43 (61.4%)
BP OG/OO	7	3 (42.9%)	1 (14.3%)	4 (57.1%)

Table 9: **Evaluator alignment with recorded debate outcomes.** We compare the evaluator’s average score across six dimensions for the original dataset-recorded winner and loser. *Gold winner scored higher* denotes strict alignment, while *gold winner higher or tied* additionally includes equal-score cases. Dataset outcomes and evaluator scores need not agree because they are produced under different evaluation criteria.

example, in this debate, the retrieved principles focused on vague or anecdotal evidence, which was a real weakness, but the central problem was burden framing and moral weighing: the debater needed to explain why violent revolution could be just despite costs, alternatives, and failed outcomes. Because the retrieved feedback did not fully target

that issue, our gains were weak. A plain LLM baseline, however, was not confounded by retrieved principles and provided generic advice, which resulted in stronger improvement. This tradeoff is acceptable, since most debates do conform to standard principles that can be matched and retrieved easily. Furthermore, since the process of graph con-

Method	DebateArt ($n = 48$)		IQ2 ($n = 34$)		BP OG/OO ($n = 3$)	
	Strict flip	Takeover	Strict flip	Takeover	Strict flip	Takeover
ReFEEDGRAPH+specific	14/48 (29.2%)	18/48 (37.5%)	11/34 (32.4%)	13/34 (38.2%)	0/3 (0.0%)	2/3 (66.7%)
ReFEEDGRAPH	12/48 (25.0%)	17/48 (35.4%)	11/34 (32.4%)	14/34 (41.2%)	1/3 (33.3%)	2/3 (66.7%)
LLM-PLAIN	9/48 (18.8%)	12/48 (25.0%)	8/34 (23.5%)	11/34 (32.4%)	0/3 (0.0%)	1/3 (33.3%)
LLM-STRUCTURED	8/48 (16.7%)	12/48 (25.0%)	7/34 (20.6%)	11/34 (32.4%)	0/3 (0.0%)	2/3 (66.7%)
PROF-FACTOR JUDGE	11/48 (22.9%)	15/48 (31.2%)	8/34 (23.5%)	11/34 (32.4%)	0/3 (0.0%)	2/3 (66.7%)
MULTI-JUDGE FEEDBACK	11/48 (22.9%)	15/48 (31.2%)	6/34 (17.6%)	8/34 (23.5%)	0/3 (0.0%)	1/3 (33.3%)

Table 10: **Winner-flip outcomes on evaluator-aligned debates.** We restrict evaluation to cases in which the evaluator initially assigns the dataset-recorded winner a strictly higher average score than the original loser. *Strict flip* requires the revised loser to score above the original winner; *takeover* additionally includes ties. Counts and percentages are reported. BP results are descriptive because only three debates satisfy the alignment condition.

struction is not as resource-intensive as training a model, new principles can be incorporated easily. Thus, reconstructing the graph is a viable solution to ameliorate this weakness.

F Case Study Feedbacks

Case Study: DebateArt 4038 Feedback Comparison

Motion: The Bible has proven the existence of God.
Score gains: Ours + specific: +2.67; LLM plain: +0.33; LLM structured: +0.33.

Our method: graph-retrieved principles and supporting examples

- **Clarify what “proof” means in your argument**

→ (Speech 2 – claim that believers prove God).******

Problem: You say believers “prove” God, but you

→ never define “proof” or show how the cited

→ verses meet that definition. The judge will see

→ this as faith-only evidence and will not accept

→ it as satisfying the burden of proof.

Why it matters: Without a clear, objective

→ definition, your claim is circular and cannot

→ be weighed against Con’s refutation that the

→ Bible is not a reliable source.

Preserve: Keep the framing that the Bible is the

→ source of evidence and that believers have a

→ unique perspective.

Rewrite move: Add a concise definition of proof,

→ e.g., “Proof is any demonstrable evidence that

→ can be observed or inferred by a rational

→ observer.” Then immediately show how Romans

→ 1:19-20 and Hebrews 11:3 provide such evidence

→ (the existence of God is manifest in creation

→ and in the way the world is framed). This keeps

→ your strategy intact while giving the judge a

→ concrete standard to apply.
- **Anchor each scriptural claim to a specific verse**

→ and explain its evidential value (Speech 4 –

→ repeated assertion that the Bible proves God).******

Problem: You repeat “the Bible proves God”

→ without linking each claim to a verse or

→ explaining why that verse is evidence. The

→ judge will see vague, unsupported assertions.

Why it matters: The judge needs to see a direct

→ connection between evidence and claim to

→ assess the burden shift.

Preserve: Keep the use of scripture as the core

→ of your evidence.

Rewrite move: For every claim, cite the exact

→ passage (e.g., Romans 1:19-20 for the manifest

→ nature of God, Hebrews 11:3 for creation by the

→ Word) and add a brief explanation: “These

→ verses show that the existence of God is not

→ merely theological but is observable in the

→ natural world.” This satisfies the principle

→ that claims must be backed by specific examples

→ (0679_P_028) and strengthens the link between

→ evidence and claim (0679_P_013).
- **Break the circularity of “God is proven to**

→ believers, therefore proven” (Speech 6 – statement

→ that God has been proven to believers).******

Problem: The argument loops: believers have proof
↳ because the Bible says so, and the Bible is the
↳ proof. The judge will see this as a logical
↳ flaw.

Why it matters: A circular argument cannot
↳ satisfy the burden of proof and weakens your
↳ burden-shifting strategy.

Preserve: Keep the claim that believers have a
↳ form of proof.

Rewrite move: Add an independent, verifiable
↳ piece of evidence from the Bible that
↳ non-believers can assess, such as fulfilled
↳ prophecy (e.g., Isaiah 53's description of the
↳ suffering servant) or archaeological
↳ corroboration of biblical events (e.g., the
↳ existence of King David). Explain how these
↳ facts can be checked by anyone, thereby showing
↳ that the Bible contains objective evidence
↳ that supports the existence of God beyond
↳ believer-only faith.

4. ****Directly address Con's refutation that the Bible
↳ is not a reliable source (Speech 4 – failure to
↳ address Con's refutations).****

Problem: You shift the burden but never rebut
↳ Con's claim that the Bible is unreliable. The
↳ judge will see an incomplete rebuttal.

Why it matters: The judge expects a full clash;
↳ failing to counter Con's main point leaves your
↳ burden shift unsubstantiated.

Preserve: Keep the burden-shift framing that Con
↳ must prove the Bible is not a source.

Rewrite move: Explicitly state, "Con must show
↳ that the Bible is not a reliable source of
↳ evidence. I will counter by presenting three
↳ lines of support: (1) internal consistency
↳ across 1,200 years of text, (2) prophetic
↳ fulfillment verified by modern scholarship, and
↳ (3) archaeological evidence that corroborates
↳ biblical events." This satisfies the principle
↳ that you must provide substantial
↳ counterarguments (0386_P.003) and keeps your
↳ strategy focused on burden shifting.

5. ****Eliminate dismissive language and personal
↳ attacks (Speech 4 & 8 – tone is dismissive).****

Problem: Phrases like "you are a fool" and
↳ "you're a fool" undermine your credibility and
↳ may cost conduct points.

Why it matters: Judges penalize tone; a
↳ respectful approach strengthens your
↳ argument's persuasiveness.

Preserve: Keep the emphasis on faith and the
↳ Bible's authority.

Rewrite move: Replace every personal attack with
↳ a respectful rebuttal. For example, change
↳ "you are a fool" to "I respectfully disagree
↳ with your assessment." This maintains your
↳ argumentative stance while meeting the judge's
↳ expectation for decorum.

6. ****Re-organize your speech into a clear, logical
↳ structure (Speech 8 – weak logical structure).****

Problem: Your final speech jumps between points,
↳ repeats the same phrases, and lacks a coherent
↳ flow. The judge will find it hard to follow.

Why it matters: A well-structured argument is
↳ easier to evaluate and more persuasive.

Preserve: Keep the four-step strategy: (1)
↳ definition of proof, (2) biblical evidence, (3)
↳ burden shift, (4) conclusion.

Rewrite move: Draft a new outline:
- ****Intro & Definition**:** Define proof and
↳ existence.
- ****Evidence**:** Cite Romans 1:19-20, Hebrews 11:3,
↳ and a fulfilled prophecy.
- ****Burden Shift**:** State that Con must prove the
↳ Bible is unreliable.
- ****Conclusion**:** Summarize that the Bible provides
↳ verifiable evidence of God's existence. Use
↳ transitions ("first," "next," "finally") and
↳ avoid repetition. This satisfies the principle
↳ that arguments must be logically structured
↳ and easy to follow.

Baseline 1: plain LLM feedback

****Debate Feedback – Mall (Pro)****
***Motion:** "The Bible has proven the existence of God"*

1. Overall Strengths

Strength	Why it matters
Passionate delivery	Your enthusiasm is clear; ↳ it keeps the audience engaged.
Use of scripture	You consistently cite verses ↳ (Romans, Hebrews, etc.), which is essential for a ↳ Bible-based argument.
Acknowledgement of the burden	You repeatedly ↳ remind the audience that the Pro side must prove ↳ God's existence.

2. Key Weaknesses

Weakness	Impact on the argument
Circular reasoning	"The Bible proves God ↳ exists because the Bible says so." This does not ↳ satisfy the burden of proof.
Lack of objective evidence	You rely on belief ↳ and faith as evidence, which the Con side can ↳ easily reject.
Repetition & filler	Many lines are repeated ↳ verbatim ("the Bible proves...") and do not add ↳ new information.
Poor structure	Your speeches jump between ↳ points without a clear roadmap, making it hard for ↳ the audience to follow.
Failure to address the Con's points	You often ↳ respond to the Con's statements in a defensive ↳ manner rather than directly refuting them.
Unclear language	Sentences such as "the Bible ↳ proves God's existence by explaining the cause of ↳ what proves or what makes known God's existence" ↳ are confusing and do not convey a logical ↳ argument.
Limited use of external evidence	No mention of ↳ fulfilled prophecies, miracles, or historical ↳ corroboration that could strengthen the claim.

3. Specific Feedback by Speech

Speech	What worked	What needs improvement
Speech 2	Cited Romans 10:17, Hebrews 11.	Repeated "the Bible proves..."; did not explain ↳ how the verses prove God's existence.
Speech 4	Attempted to refute the Con's claim ↳ of "faith is evidence."	Still circular; no ↳ concrete evidence presented.
Speech 6	Discussed the idea that proof is ↳ personal.	Did not address the Con's argument ↳ that the Bible is not a reliable source.
Speech 8	Highlighted the problem of "trapping" ↳ the Con by using the Bible as the sole source.	Still ends with "the Bible proves..." without new ↳ evidence.

4. Recommendations for Improvement

Area	Actionable Tips
Structure	Use a clear outline: 1. **Define the burden** – Pro must show **objective** proof. 2. ↳ **Present evidence** – list specific biblical ↳ claims that can be verified (e.g., fulfilled ↳ prophecies, miracles). 3. **Show external corroboration** – archaeological or historical ↳ data that supports biblical events. 4. **Address the Con's counter-arguments** – directly refute ↳ each point raised.

```

| **Evidence** | Cite specific fulfilled prophecies
↳ (e.g., Isaiah 53, Psalm 22). Reference miracles
↳ (e.g., the resurrection, the parting of the Red
↳ Sea). Bring in historical/archaeological findings
↳ that corroborate biblical narratives. |
| **Logical Clarity** | Avoid circular statements. Use
↳ "if-then" logic: "If the Bible claims X, and X is
↳ verified by Y, then the claim is credible." |
| **Conciseness** | Eliminate repeated phrases. Aim
↳ for 1-2 key points per paragraph. |
| **Engagement** | Use rhetorical questions to
↳ challenge the Con ("If the Bible is the Word of
↳ God, why does it contain contradictions?").
↳ Address the audience directly ("What would you say
↳ if...?"). |
| **Language** | Replace vague terms ("the Bible
↳ proves...") with specific claims ("the Bible's
↳ account of the crucifixion matches contemporary
↳ Roman records"). Proofread for typos and
↳ grammatical errors to maintain credibility. |
| **Handling the Con** | When the Con cites a verse,
↳ quote it, then explain *why* that verse is
↳ evidence (e.g., "Romans 10:17 shows that hearing
↳ the word leads to faith; the word itself is a
↳ divine revelation."). If the Con argues that faith
↳ is not evidence, counter with the idea that faith
↳ is the *response* to evidence, not the evidence
↳ itself. |

```

5. Quick "One-Minute" Revision Plan

```

1. **Start with a clear thesis**: "The Bible proves
↳ God's existence because it contains verifiable
↳ claims that are supported by external evidence."
2. **List 3 concrete examples** (fulfilled prophecy,
↳ miracle, historical corroboration).
3. **Show how each example meets the burden**: explain
↳ the claim, the evidence, and the logical link.
4. **Address the Con's main objection**: "Faith alone
↳ is not proof; the Bible's claims are *verified*."
5. **Conclude with a strong, memorable statement**:
↳ "When the Bible's words align with history and
↳ science, the claim of God's existence is no longer
↳ a matter of faith alone."

```

6. Final Thought

```

Your passion and scriptural knowledge are solid
↳ foundations. The challenge is to translate that
↳ into a *structured, evidence-based* argument that
↳ satisfies the burden of proof. By tightening your
↳ structure, providing concrete evidence, and
↳ directly addressing the Con's points, you'll move
↳ from a heartfelt appeal to a persuasive,
↳ rule-compliant performance. Good luck!

```

G Prompts

The main prompt templates used in the pipeline are shown below. Each box reports the stable instruction text; variables such as <DEBATE_TEXT>, <JUDGE_GUIDE>, and <RETRIEVED_PRINCIPLES> are populated at runtime.

Prompt 1: Framework-specific judge feedback

System:
You are a debate judge. Use the judge guide below as your judging philosophy.
Only evaluate the debate material provided by the user. Do not invent facts.

JUDGE GUIDE:
<JUDGE_GUIDE>

User:
Evaluate the following debate using your assigned guide.

Return ONLY valid JSON with this shape:

```

{
  "winner": "the side/team/debater you think won",
  "loser": "the side/team/debater you think lost",
  "decision": "detailed explanation of why the winner beat
the loser under this judge guide",
  "feedback_to_winner": {
    "use_the_judge_guides_recommended_section_headings":
      "give feedback to the winner using this guide's
own output format"
  },
  "feedback_to_loser": {
    "use_the_judge_guides_recommended_section_headings":
      "give feedback to the loser using this guide's
own output format"
  }
}

```

Requirements:

- Read the entire debate before judging.
- Follow the judge guide's own values, priorities, and recommended output format.
- Use the guide's section headings when available; otherwise use the guide's own criteria.
- Do not make all judges sound the same: NSDA emphasizes evidence, clash, and weighing; Toulmin emphasizes claim, grounds, warrant, backing, qualifier, and rebuttal; Rogerian emphasizes fairness, common ground, and constructive resolution; Walton emphasizes argument schemes and critical questions; UADC emphasizes burdens, clashes, persuasiveness, responses, and weighing.
- Compare central arguments through that judge guide's framework, not a generic rubric.
- Discuss specific arguments, concessions, missing responses, or weighing failures.
- Explain why the winner's arguments beat the loser's best arguments under this guide.
- Give separate, useful feedback to the winner and loser.
- Avoid generic advice unless it names the exact argument and how to improve it.
- If you cannot point to a specific argument in the debate, do not include that feedback.

DEBATE:
<DEBATE_TEXT>

Prompt 2: Sentence-level feedback classification

System:
You classify judge feedback sentences.
The input feedback is detailed and guide-specific. Do not synthesize, merge, rewrite,

abstract, or improve it yet. Preserve the judge feedback at sentence level and simply classify each substantive sentence as summary, weakness, or actionable_feedback. Return only valid JSON.

User:

Given the debate and guide-specific judge feedback from step 1, go judge by judge.

Take every substantive sentence present in each judge's feedback, including sentences inside sections like Contentions, Evidence, Clash, Warrant, Common ground, Critical questions, Weighing, feedback_to_winner, feedback_to_loser, and decision.

Classify each sentence into exactly one category:

1. summary_of_points_made_by_debaters: neutral descriptions of arguments or clashes.
2. weaknesses: evaluative statements about flaws, missing reasoning, weak evidence, poor clash, unanswered objections, tone problems, dropped arguments, or bad warrants.
3. actionable_feedback: concrete advice about what a side/debater should do differently.

Return JSON with:

- judge_name, winner, loser
- classified_sentences, each with section, target, sentence, category, reason_for_category, and debate_argument_anchor
- filtered copies for summary_of_points_made_by_debaters, weaknesses, and actionable_feedback.

Rules:

- Process every judge separately.
- Preserve section provenance and concrete debate details.
- Skip only empty fragments, section titles, or pure metadata.
- A sentence can be lightly cleaned, but do not rewrite its meaning.
- Infer debate_argument_anchor when the sentence points to a concrete argument or clash.
- If one sentence contains both weakness and advice, split only when the split is natural; otherwise choose the dominant category.
- Do not add items to category arrays that are not in classified_sentences.

DEBATE:

<DEBATE_TEXT>

JUDGE_FEEDBACK_JSON:

<JUDGE_FEEDBACK_JSON>

- Do not merge items that would require different fixes from the debater.
- Preserve source information for every merged item.

Return JSON:

```
{
  "unique_items": [
    {
      "canonical_sentence": "best concise version of the
        duplicated/unique sentence",
      "debate_argument_anchor": "specific argument/clash,
        or null",
      "merged_from": [
        {
          "sentence_id": "stable source sentence id",
          "judge_name": "judge name",
          "section": "source section",
          "target": "winner|loser|both|overall|unknown",
          "sentence": "original classified sentence"
        }
      ]
    }
  ]
}
```

Rules:

- Use stronger deduplication than exact matching.
- Merge close paraphrases and overlapping feedback.
- The canonical sentence may combine compatible details but must not add new feedback.
- For weaknesses, merge if the same debate problem is identified.
- For actionable_feedback, merge if the same fix is recommended.
- Do not merge different failures, such as tone problems and lack of evidence.

CLASSIFIED_<CATEGORY>_JSON:

<CLASSIFIED_FEEDBACK_JSON>

Prompt 3: Deduplicating classified feedback

System:

You deduplicate one category of classified debate feedback. Cluster semantically similar feedback across judges or sections and produce a strong representative item for each cluster. Preserve specificity and source traceability. Return only valid JSON.

User:

Deduplicate this category of classified feedback:
<CATEGORY>

Input:

A compact list of classified feedback sentences from the same category. Each row has sentence_id, judge_name, section, target, category, sentence, and debate_argument_anchor.

Task:

- Remove duplicates and semantically similar items.
- Merge items that express the same underlying feedback even when wording differs.
- Merge items when they point to the same skill failure or improvement area.

Prompt 4: Abstracting feedback into reusable principles

System:

You convert concrete debate feedback into reusable debate-improvement principles.

Each weakness or actionable-feedback item should become one transferable, operational principle. The principles must be general enough to apply to future debates, but specific enough to tell a debater exactly what to look for and how to repair it.

Do not use summary-only items. Return only valid JSON.

User:

Convert every item in weakness and actionable_feedback into a reusable principle. Ignore summary items.

Return JSON:

```
{
  "principles": [
    {
      "instance_feedback": "...",
      "source_specific_context": "...",
      "source_argument_anchor": "...",
      "abstract_principle": "...",
      "failure_pattern": "...",
      "trigger_pattern": "...",
      "applies_when": "...",
      "diagnostic_question": "...",
      "action_template": "...",
      "repair_steps": "...",
      "example_application": "...",
      "principle_type": "..."
    }
  ]
}
```

Rules:

- Create one principle for every weakness and actionable_feedback item.

- Preserve source specificity before generalizing; the output should read like a reusable memory card, not a slogan.
- instance_feedback should be concrete and tied to the source debate.
- source_specific_context should explain the original debate situation in enough detail that a future prompt can learn from it.
- Generalize beyond the source debate, but do not become vague.
- abstract_principle should be a conditional rule: "When X appears, do Y because Z."
- trigger_pattern should name observable signals that should cause retrieval.
- diagnostic_question should be answerable by pointing to a sentence, paragraph, source, clash, or comparison.
- repair_steps should give 4 to 6 operational steps.
- example_application should be hypothetical and reusable.
- Do not invent new debate facts.

DEDUPED_WEAKNESS_AND_ACTIONABLE_FEEDBACK_JSON:
<DEDUPED_FEEDBACK_JSON>

Prompt 5: Diagnosing issues for retrieval

System:

You identify concrete issues in debate speeches. Return only valid JSON.

User:

List concrete issues with the losing side/debater:

<LOSING_SIDE>.

Focus on specific argument failures, missing responses, weak evidence, poor weighing, tone problems, and unclear definitions.

Return JSON:

```
{
  "issues": [
    {
      "issue": "specific issue",
      "debate_argument_anchor": "specific argument/clash
        this issue concerns"
    }
  ]
}
```

DEBATE:

<DEBATE_TEXT>

Prompt 6: Graph-conditioned feedback generation

System:

You convert abstract debate-improvement principles into concrete feedback for one debate.

Return feedback only, not JSON. Be concrete, surgical, and tied to the debate text.

User:

Write specific feedback only to the losing side/debater: <LOSING_SIDE>.

Use the generic issues and retrieved principles below, but make the feedback concrete to THIS debate. If a principle says "add evidence" or "compare arguments", name the specific argument/clash where that should happen.

Strategic constraints:

- Infer the losing side's strongest existing strategy and strengthen it instead of replacing it.
- Identify 2 to 4 things the losing side already does well and tell the rewriter to preserve and sharpen them.
- Do not ask the losing side to accept the opponent's framing, burden, definitions, or evidentiary standard unless strategically necessary.

- Do not create new burdens for the losing side.
- Prefer direct clash, clearer warranting, better evidence standards, sharper burden framing, and explicit weighing.
- Treat retrieved graph principles as candidate memories, not obligations.
- Rank feedback by likely effect on the judge's decision.
- Prefer feedback likely to improve evidence, specificity, direct refutation, and relevance.

Output requirements:

- Give 4 to 7 numbered feedback items.
- Each item must name a concrete section, argument, claim, source, definition, or clash.
- Each item must name the opponent claim or judge concern it needs to answer when relevant.
- Each item must include: Problem in this debate; Why it matters; Preserve/strengthen; Rewrite move.
- Use retrieved principles as repair rules, not generic advice.
- Do not say only "add evidence", "be clearer", "engage more", or "improve structure".
- For each rewrite move, explain how it strengthens the losing side's existing route to winning.
- Do not copy old supporting examples as if they happened in this debate.

GENERIC ISSUES:

<GENERIC_ISSUES_JSON>

RETRIEVED PRINCIPLES:

<RETRIEVED_PRINCIPLES_JSON>

SUPPORTING SPECIFIC FEEDBACK EXAMPLES:

<SUPPORTING_EXAMPLES_JSON>

DEBATE:

<DEBATE_TEXT>

Prompt 7: Baseline direct LLM feedback

System:

You are a helpful debate feedback writer.

User:

Give feedback to this debater: <LOSING_SIDE>.

DEBATE:

<DEBATE_TEXT>

Prompt 8: PROF-factor feedback baseline

System:

You are a debate judge focused on revision-useful feedback.

User:

You are a debate judge giving feedback to this debater: <LOSING_SIDE>.

Judge their performance using these PROF-style debate revision factors:

- persuasiveness: whether the speech would convince a debate judge.
- eloquence: clarity, organization, tone, and fluency.
- evidence: quality, relevance, and integration of evidence or examples.
- correctness: factual accuracy, logical validity, and fair treatment of the opponent.
- relevance: focus on the motion, burdens, and live clashes in the debate.
- specificity: use of concrete arguments, named clashes, and exact opponent responses.

Give feedback that would help <LOSING_SIDE> revise their existing debate speeches.

Refer to specific arguments and clashes from the transcript. Do not invent new evidence

or tell the debater to add content unrelated to their original position.

DEBATE:
<DEBATE_TEXT>

Prompt 9: Revision model

System:

You are revising one side's debate speech in response to feedback.
Use the full debate transcript as context, but rewrite only the losing side's speech.
Your goal is to produce a stronger revised version of that side's speech, not a judge explanation and not feedback.
You must be conservative: revise only the parts of the original speech targeted by the feedback, and preserve all other content.

User:

Debate motion:
<TITLE>

Losing side/debater to revise:
<LOSING_SIDE>

Task:

- Rewrite only <LOSING_SIDE>'s speech.
- Use the full transcript only to apply the feedback, not to invent a new case.
- Apply every substantive feedback item, but only by editing the relevant part.
- Addressability gate: only make a substantive revision when the feedback gives a concrete target or concrete edit.
- If feedback is vague, do not invent the missing instruction.
- Do not add new arguments, sources, statistics, examples, burdens, or framing unless explicitly requested and grounded in the original speech or transcript.
- Keep unchanged sections unchanged when the feedback does not target them.
- Preserve the original position, organization, evidence, citations, examples, and voice.
- When feedback says to answer an opponent claim, add the answer in the relevant section.
- When feedback says to define a term, add the definition where that term already matters.
- When feedback says to support a claim, add reasoning from the debate record or explain what the cited source/argument establishes. Do not invent fake citations.
- Remove or soften dismissive language when it weakens credibility.
- Output only the revised speech.

FULL DEBATE TRANSCRIPT:
<DEBATE_TEXT>

ORIGINAL SPEECH TO REVISE:
<ORIGINAL_SPEECH>

FEEDBACK TO APPLY:
<FEEDBACK>

Debate motion:
<TITLE>

Use the full debate transcript only as context for the opponent's claims, burdens, definitions, evidence, concessions, and clashes. Score only the TARGET SPEECH.

Return JSON with exactly these keys. Each score must be a number from 0 to 5:

```
{
  "persuasiveness": {"score": 0, "reason": "reason"},
  "eloquence": {"score": 0, "reason": "reason"},
  "evidence": {"score": 0, "reason": "reason"},
  "correctness": {"score": 0, "reason": "reason"},
  "relevance": {"score": 0, "reason": "reason"},
  "specificity": {"score": 0, "reason": "reason"}
}
```

Rubric:

- persuasiveness: likelihood of moving a judge, considering burdens, framing, impact, weighing, and answer quality.
- eloquence: clarity, organization, signposting, readability, and professional tone.
- evidence: quality, relevance, integration, explanation, source use, and comparison.
- correctness: logical validity, consistency, fair representation, and avoidance of fallacies.
- relevance: engagement with the motion, live clashes, burdens, and voting issues.
- specificity: concrete claims, named arguments, sources, definitions, clash references, and paragraph-level detail.

FULL DEBATE TRANSCRIPT:
<DEBATE_TEXT>

TARGET SPEECH:
<TARGET_SPEECH>

Prompt 10: Revision evaluator

System:

You are a debate judge evaluating one side's speech in the context of the full debate.
Score the target speech as debate performance, not as a generic essay.
Return only valid JSON. Do not use Markdown, code fences, comments, or prose outside the JSON object.

User:

Evaluate the TARGET SPEECH for side/debater: <SIDE_NAME>