# Generating Critical Questions for Arguments

## A Large-Scale Analysis of Prompting and Reranking Strategies

**Midhun Kanadan**
Bauhaus-Universität Weimar
midhun.kanadan@uni-weimar.de

**Benno Stein**
Bauhaus-Universität Weimar
benno.stein@uni-weimar.de

## Abstract

Critical Question Generation aims to create questions that challenge an argument's assumptions, evidence, or reasoning. But what exactly does "challenge" mean? This paper first discusses the origins of the idea of critical questions and proposes a definition that accounts for the *audience dependence* of arguments. Its main contribution is a large-scale empirical study on the CQs-Gen benchmark, covering 42 language models, 18 prompting strategies, supervised fine-tuning, and a generate-then-select approach. We find that our proposed architecture with a fine-tuned ModernBERT reranker achieves 82.4% on the CQs-Gen evaluation score, surpassing the winner of the ArgMining 2025 shared task by 14.8 percentage points. We test the prevailing assumption that criticality can be derived from argumentation scheme templates and show this assumption to be incomplete: scheme-aware prompting can even degrade performance (by around 4% in 23 of 34 open-weight models, with a directionally consistent but weaker effect in proprietary systems). We also observe that compact fine-tuned models surpass larger zero-shot systems. Together, these results suggest that generating practically useful critical questions benefits from context-sensitive generation and selection rather than scheme-template matching.

## 1 Introduction

Consider the following argument:

> "Students can now obtain instant and accurate answers from artificial intelligence systems; therefore, AI will soon replace teachers."

Before accepting the conclusion, a critical thinker might ask: Does providing instant answers substitute for the pedagogical value of human guidance and mentorship? And does answering factual questions generalize to the full range of learning outcomes that education aims to produce? Such questions are examples of critical questions.

Loosely speaking, critical questions challenge the assumptions, evidence, or line of reasoning underlying an argument. In argumentation theory, critical questions are traditionally associated with argumentation schemes, recurring patterns of reasoning such as arguments from expert opinion, analogy, or cause to effect (Walton et al., 2008). Each scheme is accompanied by canonical questions intended to test whether the argument should be accepted (Walton et al., 2008). This scheme-based perspective has strongly influenced recent work in computational argumentation and the emerging task of Critical Question Generation (CQG), where systems automatically generate questions that probe the weaknesses of argumentative texts (Calvo Figueras and Agerri, 2024; Calvo Figueras et al., 2025). Unlike standard question generation, CQG requires identifying what an argument takes for granted rather than completing or responding to it.

CQG has broad practical value. Consider a judge evaluating a legal argument, a doctor reviewing a medical claim, or an analyst assessing a political debate: in each case, asking the right critical question at the right moment could change the outcome entirely. Beyond these high-stakes settings, critical questions enable misinformation prebunking by immunizing people against weak arguments before they take hold, rather than correcting falsehoods after they have spread (Roozenbeek et al., 2022; Musi et al., 2023), and surface the implicit assumptions in argumentative texts for essay evaluation (Song et al., 2014) and deliberative policy-making. These applications share a common requirement: questions must be generated at scale, across the continuous stream of argumentative content encountered daily. Automation is not a convenience but a necessity.

The dominant approach to CQG draws on argu-



*Proceedings of the 19th International Natural Language Generation Conference: Long Papers*, pages 601–622
October 17-21, 2026 ©2026 Association for Computational Linguistics

mentation theory and assumes that criticality can be derived from argumentation scheme templates: identify the scheme, then instantiate its associated canonical critical questions. This assumption faces two practical challenges: reliable scheme identification is required, yet automatic scheme classification remains challenging for language models (Kondo et al., 2021; Bezou-Vrakatseli et al., 2025), and even when correctly applied, template-derived questions tend to be too generic to meaningfully challenge a specific argument. In practice, critical questioning appears to arise from contextual uncertainties about a specific argument rather than from the retrieval of scheme-specific templates.

Large language models offer a natural alternative, but fluency does not translate to critical precision: LLMs tend to produce generic outputs that do not challenge the specific assumptions of the argument at hand, and can hallucinate by introducing concepts absent from the text. Injecting scheme information into prompts constrains the model's generative flexibility, producing questions that are topically relevant but not genuinely critical. Effective CQG therefore requires structured prompting tailored to the argument and a principled selection mechanism.

We demonstrate that context-sensitive generation and selection offers a more effective path to CQG than scheme-template matching, through persona-driven prompting, supervised fine-tuning of compact open-weight models, and generate-then-select architectures with discriminative reranking.

This paper makes four contributions. First, it proposes an observer-relative definition of critical questions that generalizes beyond scheme-template formulations. Second, it presents the first large-scale empirical characterization of CQG across prompting, decoding, fine-tuning, and reranking settings, showing that scheme-aware prompting degrades performance both when using open-weight models and proprietary systems. Third, it shows that supervised fine-tuning enables compact open-weight models to surpass substantially larger zero-shot systems, demonstrating a practical deployment path independent of proprietary APIs or server-grade infrastructure. Finally, it establishes a new state of the art on the CQs-Gen benchmark through generate-then-select architectures with discriminative reranking, surpassing the ArgMining 2025 shared task winner by 14.8 usefulness percentage points.[1]

Section 2 surveys related work on argumentation theory, CQG, and reranking methods. Section 3 introduces the observer-relative working definition. Section 4 describes the task, dataset, and experimental setup for prompt-based generation, supervised fine-tuning, and generate-then-select reranking. Section 5 presents results and analysis across all three paradigms. Section 6 concludes.

## 2 Related Work

Critical questions originate in classical argumentation theory, with roots extending from Aristotle's *Topics* (Aristotle, 1960) to Hamblin's reframing of fallacies as violations of dialogue rules (Hamblin, 1970). Hastings (1963) introduced the modern critical question device by pairing each argumentation scheme with diagnostic queries targeting defeasible premises, later operationalized by Walton (1996) and Walton et al. (2008) into approximately sixty named schemes with canonical critical question sets. Despite their usefulness for analyzing argument structure, argumentation schemes do not constitute a complete evaluation framework, particularly in practical argumentative settings.

Several limitations of scheme-based critical question definitions have since been identified. Hernández (2023) points out two main problems. First, no finite set of scheme-level questions can exhaust the evaluation conditions for a concrete argument (the completeness problem). Second, what counts as critical depends on the interlocutor's existing knowledge and assumptions (the observer-blindness problem). Walton and Godden (2005) further argued that critical questions function dialogically as mechanisms for challenging presumptive arguments rather than merely as static structural tests. A further computational complication is that scheme classification remains unreliable across language models (Kondo et al., 2021; Bezou-Vrakatseli et al., 2025), and many real-world arguments may instantiate multiple overlapping schemes.

Calvo Figueras and Agerri (2024) introduced Critical Question Generation (CQG) as a formal NLP task, distinguishing it from reading-comprehension question generation and counter-argument generation. Calvo Figueras and Agerri (2025) constructed the CQs-Gen dataset of 5K

[1] https://huggingface.co/spaces/HiTZ/Critical_Questions_Leaderboard

manually annotated questions with a three-way label scheme (*Useful*, *Unhelpful*, and *Invalid*) evaluated by journalists and fact-checkers, and showed that reference-based semantic similarity metrics correlate more strongly with human judgments than direct classification approaches. The ArgMining 2025 shared task on CQG (Calvo Figueras et al., 2025) benchmarked thirteen systems, with the top system achieving 67.6% on the evaluation score. The shared-task results suggest that model selection has greater impact than prompt engineering, while classification and ranking modules consistently improve output quality. Prior systems further suggested that argumentation schemes may help when integrated at a selection stage, whereas direct scheme-constrained prompting often produces generic outputs.

Recent work has also incorporated critical questions into LLM reasoning pipelines for complex reasoning tasks, highlighting the broader applicability of critical questioning beyond argumentation analysis (Castagna et al., 2024). A related encoder-based reranking approach (Kanadan et al., 2025) used a binary classifier. This work instead employs a three-class reranker, alongside a large-scale generation analysis and an observer-relative theoretical account.

The generate-then-select paradigm, in which a diverse candidate pool is generated and a secondary model selects the most suitable output, is well established in NLP. Encoder-based cross-encoders have shown strong reranking performance in information retrieval (Nogueira and Cho, 2019), and ModernBERT (Warner et al., 2024) extends this to sequences of up to 8,192 tokens, making it well suited to long argumentative interventions. An alternative is LLM-as-a-judge (Zheng et al., 2023), but LLM judges exhibit systematic positional bias, length bias, and self-preference (Wang et al., 2024), which can undermine reliability in quality-sensitive selection tasks. Both paradigms are evaluated in this paper in the context of CQG.

## 3 How to Define Critical Questions?

Critical questions have been defined from several theoretical perspectives. In the informal logic tradition, Walton (1996) and Walton et al. (2008) treat critical questions as part of each argumentation scheme's structure, representing typical ways an argument can be attacked, with satisfactory answers constituting a necessary condition for argument acceptability (Walton and Godden, 2005). In the NLP context, Calvo Figueras and Agerri (2024) operationalize critical questions as tools that expose the blind spots of an argument by pointing at information it could be missing, and the ArgMining 2025 CQG shared task (Calvo Figueras et al., 2025) defines them as inquiries designed to unmask the assumptions held by an argument's premises and challenge its inferential structure.

Existing scheme-based definitions exhibit important theoretical and practical limitations. First, no finite set of scheme-level template questions can exhaust all evaluation conditions for concrete arguments. Real arguments contain contextual assumptions and implicit reasoning steps that predefined templates cannot fully anticipate (Hernández, 2023). Second, criticality depends partly on the evaluator's existing knowledge and epistemic commitments (what Hernández (2023) terms the observer-blindness problem). A question that substantially challenges one observer may be entirely uninformative for another who already accepts the relevant premise. Under scheme-based definitions, however, criticality is treated primarily as a structural property of the argument type itself. Structural validity and practical criticality are therefore related but distinct properties.

A further limitation is the dependency on argumentation scheme classification that scheme-based approaches require. To apply scheme-specific critical questions, a system must first identify the underlying scheme instantiated by the argument. However, automatic scheme classification remains challenging for language models, and real-world arguments frequently combine multiple overlapping reasoning patterns. Moreover, human critical questioning rarely proceeds through explicit scheme identification: argumentative discourse is often incomplete, ambiguous, and context-dependent, making rigid scheme-template mappings difficult to apply reliably.

These limitations motivate the following definition.

**Definition 1** (Observer-Relative Critical Question). *Given an argument $A$ and an observer $O$, a question $q$ is a critical question for $A$ with respect to $O$ when the answer to $q$ changes $O$'s belief in $A$.*

The observer $O$ may be a human or an AI system, and stronger critical questions are those whose answers produce larger changes in the observer's belief that the argument holds.

Under this definition, criticality is a relational property involving the argument, the question, and the observer, rather than a structural feature of the argument alone.[2] Scheme alignment alone does not determine practical criticality, but remains one route by which a question may achieve it.

Whether the answer to a question can shift belief in an argument depends on what the observer already knows and accepts: their familiarity with the topic and its terminology, domain expertise, prior skepticism toward the claim, and the commitments they bring to the argument, such as ideological alignment, group loyalty, or a personal stake in its conclusion. For example, a question asking how a certain term in the argument is defined does no work as a critical question for a reader (observer) who already knows the meaning of the term, but can reveal to an observer unfamiliar with this term that the argument rests on an assumption the observer had not examined, once that meaning is made explicit (i.e., the question is answered). Likewise, a question probing the credentials of a cited authority is critical for a reader unfamiliar with that authority and inert for one who already knows them well. Criticality is therefore not a fixed attribute of a question, and any evaluation of critical questions is an evaluation relative to some observer, whether or not that observer is made explicit.

For the experiments in this paper, the observer is fixed and implicitly encoded in the ground truth of the benchmark dataset. The annotation guidelines underlying this ground truth are described in Section 4.1. In Appendix A, we analyze these guidelines in detail and show that they themselves presuppose observer-relativity: rather than eliminating it, the annotation process instantiates, and thereby fixes, a particular observer perspective in the resulting ground truth. This matters for the interpretation of the benchmark, since questions labeled Unhelpful or Invalid under this standard may nonetheless be critical for observers with different backgrounds, knowledge, or commitments.

Finally, measurement of belief change and systematic variation of the observer across populations are left to future work and discussed in Section 6.

[2]Minsky 1965 makes an analogous observation about models: the model relation is "inherently ternary," and suppressing the observer "leads to circular definitions." Suppressing the observer from a definition of criticality generates the same problem.

## 4 Generating Critical Questions with LLMs

Three generation paradigms are examined to identify the most effective approach to CQG: prompt-based generation, which requires no parameter updates; supervised fine-tuning, which adapts model weights to task-specific examples; and generate-then-select architectures, which improve quality through candidate diversity and discriminative reranking. All three are evaluated on the CQs-Gen benchmark under a unified task formulation and evaluation protocol, forming the basis of the empirical study.

Three research questions guide the empirical study: (1) How do model family, scale, prompting strategy, and decoding configuration affect CQG quality, particularly under scheme-aware prompting? (2) Can supervised fine-tuning enable compact open-weight models to surpass larger zero-shot systems at substantially lower parameter counts? (3) Do generate-then-select architectures improve over direct generation, and how do encoder-based and LLM-based rerankers compare?

### 4.1 Task and Dataset

To characterize how LLMs can be best configured for CQG, a systematic evaluation is conducted on the CQs-Gen benchmark, the first large-scale resource for this task (Calvo Figueras and Agerri, 2025). The benchmark was introduced through the ArgMining 2025 shared task on Critical Question Generation.[3] Thirteen teams participated, proposing systems based on prompt engineering, retrieval-augmented generation, and reranking; a full description of all submitted systems is provided in the shared task overview (Calvo Figueras et al., 2025). A related encoder-based reranking approach (Kanadan et al., 2025) achieved 56.9%, ranking fifth in the shared task.

**Task** Given an argumentative intervention, the task is to generate exactly three critical questions that a competent critical thinking evaluator would consider useful for assessing the argument's reasoning, without requiring external factual knowledge, focusing evaluation on reasoning quality rather than factual verification.

**Dataset** The CQs-Gen benchmark is constructed from four English argument mining corpora:

[3]https://hitz-zentroa.github.io/shared-task-critical-questions-generation/

Table 1: CQs-Gen dataset statistics. The test set has a more balanced and harder label distribution than the training set by design, with a substantially higher proportion of Unhelpful and Invalid reference questions.

| Split | Argument. intervent. | Critical questions | Useful | Unhelpful | Invalid |
|---|---|---|---|---|---|
| Training | 186 | 4,136 | 67.5% | 21.6% | 10.9% |
| Test | 34 | 806 | 42.7% | 31.0% | 26.3% |

US2016 presidential debate transcripts (Visser et al., 2021), BBC Moral Maze debates (Lawrence et al., 2018), US2016reddit, and the Regulation Room Divisiveness corpus (Konat et al., 2016). The dataset comprises 220 argumentative interventions annotated with one or more argumentation schemes from the Walton taxonomy. Reference critical questions were generated by two complementary methods: instantiating Walton-style scheme templates for each intervention, and prompting Llama-3-70B-Instruct in a zero-shot setting. The resulting questions were manually annotated by journalists and fact-checkers under a three-way scheme: *Useful* (challenges an assumption or evidential gap that, if unanswered, weakens the argument), *Unhelpful* (topically relevant but generic, or already answered in the text), and *Invalid* (off-topic, hallucinated, or not critical). Notably, only 41.8% of Walton-style template questions are labeled Useful, compared to 75.6% for LLM-generated reference questions, indicating that scheme-derived questions are often topically valid but too generic to challenge a specific argument. Table 1 reports the dataset statistics.

**Evaluation Metric** Each generated question is compared to the annotated gold references for its intervention and inherits the label of the most similar reference, determined by cosine similarity over sentence embeddings with a threshold of 0.65;[4] questions below the threshold are marked as not-able-to-evaluate (NAE). Questions are evaluated separately and their scores aggregated: each Useful question contributes $\frac{1}{3}$ to the system's *usefulness score*, while Unhelpful and Invalid questions contribute 0, yielding a score between 0 and 1.

### 4.2 Prompt-Based Generation

Prompt-based generation provides a model-agnostic baseline for studying how LLMs interpret CQG without task-specific parameter updates, enabling systematic comparison across model families, scales, and decoding strategies.

The 18 prompting strategies span seven categories, which are summarized in Appendix H: baseline zero-shot instructions, constraint-oriented prompts with explicit quality prohibitions, persona-based prompts (Investigative Journalist, Critical Thinking Teacher; Figure 6), scaffolding prompts with intermediate reasoning steps, few-shot prompts with positive and negative examples, hybrid combinations, and scheme-aware prompts that inject Walton-style scheme definitions from the CQs-Gen annotations at inference time (Figure 7). The last category is included to test empirically whether formal argumentation structure improves criticality rather than assuming it does.

The evaluation covers 42 language models: 34 open-weight systems (270M to 32B parameters) and 8 proprietary frontier models; the full model list is provided in Appendix B. Open-weight models are evaluated under both default sampling and greedy decoding (temperature $= 0$); proprietary models use their standard API defaults. Each model generates exactly three critical questions per intervention.

### 4.3 Supervised Fine-Tuning

While prompt-based generation exposes the difficulty of producing context-specific and genuinely useful critical questions, supervised fine-tuning provides a direct alternative by adapting model behavior to the specific demands of CQG. The investigation addresses whether task-specific supervision can compensate for scale limitations, enabling compact and locally deployable models to approach or exceed the performance of substantially larger zero-shot systems.

Eight open-weight models spanning 1B to 27B parameters are fine-tuned using QLoRA (Dettmers et al., 2023), a parameter-efficient fine-tuning method suitable for resource-efficient adaptation. The training set is constructed exclusively from questions labeled Useful by human annotators in the training split of CQs-Gen, yielding 979 training examples from 186 interventions. Rather than learning from the full distribution of generated questions, including Unhelpful and Invalid cases, models are trained specifically on examples judged effective at challenging argumentative reasoning. Only open-weight models are considered, as closed-source systems do not expose weights

[4]The official shared task evaluation protocol uses the `stsb-mpnet-base-v2` sentence embedding model (Reimers and Gurevych, 2019).

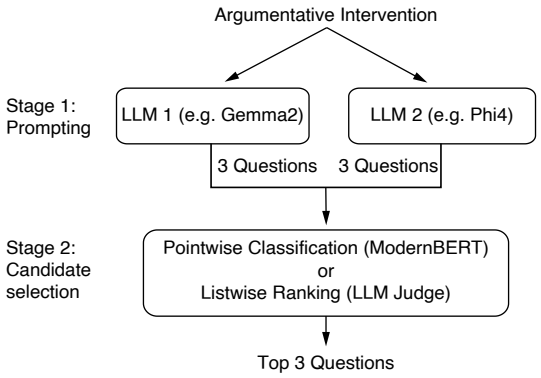



Figure 1: The two-stage generate-then-select pipeline for CQG. Two generator LLMs each produce three candidate critical questions per argumentative intervention, yielding a pool of six, a pointwise classifier (ModernBERT) or listwise LLM judge selects the final three.

for supervised adaptation. Full model lists, training configuration, instruction templates, and implementation details are provided in Appendix D.

### 4.4 Generate-Then-Select Approach

In the generate-then-select paradigm, two generator LLMs each produce three candidate questions per intervention, yielding a pool of six candidates. Because usefulness is highly context-dependent, generating multiple diverse candidates and selecting among them is more reliable than depending on a single generation trajectory. A secondary model therefore selects the top three according to estimated usefulness. Figure 1 illustrates the two-stage pipeline. Two selection strategies are evaluated.

The first strategy uses encoder-based pointwise classification. ModernBERT (Warner et al., 2024) is fine-tuned as a three-class classifier (Useful, Unhelpful, Invalid). ModernBERT's 8,192-token context window ensures that the full intervention and candidate question are encoded without truncation. The classifier is trained on approximately 56,500 examples, combining gold training annotations with LLM-generated candidate questions whose labels are inherited from the nearest matching gold reference, to compensate for the limited size and class skew of the gold set alone; full training and selection details are provided in Appendix E.

The second strategy uses listwise LLM-based ranking. Unlike pointwise classification, listwise evaluation allows the judge to assess how well the selected set collectively challenges different aspects of the argument, avoiding redundant candidates that target the same weakness. Gemini 2.5 Pro is used as the judge after preliminary experiments showed that open-weight LLMs struggled with reliable candidate discrimination and consistent output formatting. The judge evaluates all six candidates jointly and selects the three that provide the strongest combined critique of the argument. Both strategies operationalize the observer-relative criterion from Section 3: each reranker approximates a competent critical thinker, with ModernBERT calibrated to human annotator judgments and the LLM judge using an investigative journalist persona.

Table 2: Top-10 models on the CQs-Gen test set, ranked by average usefulness score across 18 prompts. 'Best' is the strongest single-prompt result, NAE is the proportion of not-able-to-evaluate outputs, averaged across prompts. An asterisk (*) marks open-weight models under greedy decoding. **Bold**: best average, **shaded**: best single result, underline: best open-weight average.

| Model | Usefulness Score (Avg.) | Usefulness Score (Best) | NAE (%) |
|---|---|---|---|
| gpt-4.1 | **0.550** $\pm$ 0.048 | 0.647 | 10.4 |
| grok-4-0709 | 0.542 $\pm$ 0.039 | 0.598 | 12.9 |
| grok-4-fast-reasoning | 0.536 $\pm$ 0.028 | 0.608 | 11.9 |
| gemma2-27b* | 0.530 $\pm$ 0.043 | **0.657** | 10.9 |
| gemini-2.5-flash | 0.522 $\pm$ 0.066 | 0.627 | 16.0 |
| phi4-14b* | 0.521 $\pm$ 0.038 | 0.588 | 9.8 |
| qwen2.5-32b* | 0.517 $\pm$ 0.064 | 0.647 | 10.1 |
| mistral-7b* | 0.513 $\pm$ 0.049 | 0.627 | 7.5 |
| gpt-4o | 0.510 $\pm$ 0.048 | 0.588 | 10.4 |
| gemma2-9b* | 0.505 $\pm$ 0.039 | 0.578 | 9.6 |

## 5 Results and Analysis

Results are reported in turn for each of the three generation paradigms, followed by a cross-paradigm analysis.

### 5.1 Prompt-Based Generation

Table 2 reports the top results from the prompt-based generation study. The highest *average* score across prompts belongs to GPT-4.1 (0.550), but the best single-prompt result belongs to Gemma 2 27B with Prompt 16 (Investigative Journalist persona) under greedy decoding (0.657), reflecting greater prompt sensitivity in open-weight models.

Scheme-aware prompting degrades performance in 23 of 34 open-weight models, with a mean degradation of $-4.15\%$ relative to the scheme-independent persona configuration. The degradation is concentrated in the Unhelpful category rather than the Invalid category: scheme-aware prompting shifts questions that would otherwise be

Useful toward generic but structurally valid Unhelpful outputs, consistent with the analysis of Walton-style template questions in Section 4.1. The effect is directionally consistent but weaker in proprietary systems: 6 of 8 show negative change with a mean degradation of $-0.61\%$, driven upward by a single strong positive outlier (Gemini 2.5 Flash, $+14.7\%$). Stronger instruction-following capacity may partially compensate for template rigidity.

The Investigative Journalist persona prompt achieves the highest average usefulness score across models (0.462) and the best single-model result (0.657), but also exhibits the highest variance: stronger architectures extract substantial value from the persona framing while weaker models struggle to operationalize it. Simpler guideline-based prompts provide more stable results across architectures and represent a reliable lower-risk alternative for deployment. No model achieves its best result under baseline zero-shot or scheme-aware prompting. Greedy decoding provides a modest but consistent benefit for most open-weight models (23 of 34), with particularly reliable gains for the Gemma 2 and Qwen 2.5 families. Proprietary models are more consistent across prompt configurations overall, while open-weight models show wider variance but can match proprietary systems under optimal settings. Full analyses are in Appendix H; training results in Appendix C.1.

### 5.2 Supervised Fine-Tuning

Table 3 reports fine-tuning results for eight open-weight models (base scores use the fixed fine-tuning prompt, not the best-of-18 result in Table 2). Supervised adaptation improves performance in seven of eight models, with an average gain of $+0.062$ usefulness points over the zero-shot baseline for the same models, and the largest gains observed in compact models where zero-shot prompt-following is less reliable.

The results reveal a clear size-dependent pattern. Compact models (1B–3B) benefit most: fine-tuned Gemma 2 2B (0.59) surpasses Gemma 2 27B's matched-prompt zero-shot score (0.51) at one-thirteenth of the parameter count, demonstrating that task-specific supervision can compensate for scale and enabling deployment on consumer hardware without proprietary APIs or server-grade infrastructure. Mid-scale models (7B–14B) show moderate but consistent gains of $+0.02$ to $+0.07$. By contrast, fine-tuning yields negligible returns for the largest model: Gemma 2 27B regresses slightly ($\Delta = -0.01$), suggesting that large instruction-tuned models have limited headroom for further adaptation on a small task-specific dataset. Beyond the raw usefulness score, fine-tuning also improves output quality more broadly: Invalid outputs decrease by approximately 27%, NAE cases by approximately 22%, and fine-tuned models generate on average 6.4 more Useful questions per model than their base counterparts. The scaling trend is visualized in Appendix D (Figure 3).

Table 3: Supervised fine-tuning results on the CQs-Gen test set. 'Base score' is the model's zero-shot score under the fixed instruction prompt used for fine-tuning (Figure 4), not the best-of-18-prompts score reported in Table 2; this isolates the effect of fine-tuning from prompt selection. 'SFT score' is the score after fine-tuning, '$\Delta$ score' is the resulting change. '$\Delta$ Invalid' and '$\Delta$ Useful' are absolute changes in question counts across all 34 test interventions.

| Model | Base Score | SFT Score | $\Delta$ Score | $\Delta$ Invalid | $\Delta$ Useful |
|---|---|---|---|---|---|
| Llama 3.2 1B | 0.42 | 0.52 | **+0.10** | –5 | +10 |
| Qwen 3 1.7B | 0.39 | 0.50 | **+0.11** | +1 | +11 |
| Gemma 2 2B | 0.49 | 0.59 | **+0.10** | –10 | +10 |
| Llama 3.2 3B | 0.49 | 0.56 | **+0.07** | –7 | +7 |
| Mistral 7B | 0.52 | 0.57 | +0.05 | –3 | +5 |
| Gemma 2 9B | 0.54 | 0.61 | **+0.07** | –7 | +7 |
| Qwen 2.5 14B | 0.51 | 0.53 | +0.02 | –1 | +2 |
| Gemma 2 27B | 0.51 | 0.50 | –0.01 | –3 | –1 |

### 5.3 Generate-Then-Select Approach

Table 4 reports results for the generate-then-select reranking experiments. Encoder-based reranking with ModernBERT substantially outperforms both direct single-model generation and LLM-based judge reranking across all evaluated generator configurations.

The best configuration pairs Gemini 2.5 Pro and Gemma 2 27B as generators with ModernBERT as the discriminative reranker, achieving 82.4% usefulness and surpassing the ArgMining 2025 shared task winner (Favero et al., 2025) (67.6%) by 14.8 percentage points. A fully open-weight configuration pairing Gemma 2 27B and Mistral 7B achieves 81.4%, within 1.0 percentage point of the best mixed configuration, indicating that the gain stems mainly from candidate pool diversity rather than individual generator quality. Notably, scheme-aware prompts that perform poorly in single-model generation (typically below 0.50) improve to around 0.62 in the hybrid reranking

Table 4: Best generate-then-select configurations on the CQs-Gen test set. 'Usefulness Score' is the proportion of Useful questions, the baseline is Gemma 2 27B with its best prompt configuration (0.657). LLM Judge uses Gemini 2.5 Pro as a listwise selector. * indicates a fully open-weight configuration. Full top-10 results for both strategies are provided in Appendix F.

| Generator 1 | Generator 2 | Reranker | Usefulness Score | Invalid% | NAE% |
|---|---|---|---|---|---|
| Gemini 2.5 Pro | Gemma 2 27B | ModernBERT | **0.824** | 4.9 | 6.9 |
| Gemma 2 27B | Mistral 7B* | ModernBERT | 0.814 | 5.9 | 4.9 |
| DeepSeek-R1 32B | Gemma 2 27B | LLM Judge | 0.725 | 7.8 | 6.9 |
| Gemma 2 27B | – | (baseline) | 0.657 | 12.7 | 4.9 |

setting, demonstrating that the ModernBERT classifier can partially compensate for prompt-induced degradation by surfacing better candidates from a mixed pool.

ModernBERT also functions as a safety filter: the proportion of Invalid questions in the final output is reduced to approximately 7.1% on average across all encoder-based configurations evaluated (top-10 average: 5.4%), compared to approximately 11–13% for single-model generation. The NAE rate stabilizes at around 13% across encoder-based configurations. The LLM judge approach improves over the single-model baseline and still surpasses the ArgMining 2025 shared task winner (67.6%) by 4.9 percentage points, but falls 9.9 percentage points below encoder-based reranking, consistent with the systematic positional bias and label miscalibration documented in prior work (Wang et al., 2024). The LLM-judge configurations also exhibit higher Invalid rates on average ($\sim$9.2%) and greater NAE variability, with some model pairs reaching approximately 20%. These LLM judges provide natural-language justifications for each selection, making them useful as analysis tools where interpretability is valued.

### 5.4 Analysis

**Why Scheme-Aware Prompting Fails** As documented in Section 5.1, scheme-aware prompts shift Useful outputs toward Unhelpful rather than Invalid (Figure 7), confirming that templates produce structurally valid questions that nonetheless fail to challenge a competent evaluator's belief in the specific argument. For example, applying the Argument from Expert Opinion scheme to a police-authority argument yields "Is the police a genuine expert in law and order?": structurally valid and topically connected, yet too generic to challenge any specific claim in the argument and therefore not critical. This failure is consistent with the observer-blindness problem identified in Section 3: scheme templates cannot account for what a specific observer does not know, trust, or accept about the concrete argument, because they operate at the level of scheme-type abstractions rather than argument-specific dependencies.

**Why Encoder Reranking Outperforms LLM Judges** The performance advantage stems from two distinct factors. First, ModernBERT is trained directly on the CQs-Gen label distribution and scores each candidate independently, making selections invariant to candidate ordering, whereas LLM judges exhibit positional sensitivity (Wang et al., 2024). Second, ModernBERT's decision boundary is calibrated to the exact annotation guidelines, whereas zero-shot LLM judges over-predict the Useful label and produce a higher average Invalid rate (LLM judge: $\sim$9.2%; ModernBERT: $\sim$7.1%).

**Scaling and Decoding** Greedy decoding benefits 23 of 34 open-weight models (mean $+0.0026$) but the effect is strongly family-dependent: Gemma 2 and Qwen 2.5 benefit consistently, while DeepSeek-R1 models often degrade (mean absolute change $-0.0326$). Within-family scaling generally improves performance, but architecture matters beyond parameter count: Qwen 3 outperforms similarly-sized Qwen 2.5 models at intermediate scales but not at the smallest or largest configurations, while DeepSeek-R1 shows diminishing returns beyond mid-scale configurations (Figure 2 in Appendix C.2).

**Deployment Implications** The open-weight-only reranking configuration (0.814) and the compact fine-tuning result together demonstrate that high-quality CQG is achievable on local infrastructure without proprietary APIs or server-grade compute. Inspection of model outputs reveals three characteristic failure modes affecting compact open-weight models in deployment: safety-driven refusals, model collapse, and contextual star-

vation on incomplete interventions (Appendix G).

## 6 Conclusion

This paper presents the first large-scale empirical study of Critical Question Generation, examining 42 language models across 18 prompting strategies, supervised fine-tuning, and a generate-then-select approach on the CQs-Gen benchmark. Alongside this empirical investigation, an observer-relative definition of critical questions is proposed: a question is critical if its answer changes the observer's belief in the argument. This definition treats criticality as a relational property involving the argument, the question, and the observer, rather than a structural feature of the argument alone, offering a scheme-independent account consistent with the failure modes documented in the experiments.

Three main findings emerge: first, scheme-aware prompting degrades performance in 23 of 34 open-weight models (mean $-4.15\%$), with a directionally consistent but weaker effect in proprietary systems (6 of 8; mean $-0.61\%$). In both cases the degradation is concentrated in the Useful-to-Unhelpful direction, confirming that criticality depends on argument-specific context rather than scheme-level structure. Second, targeted fine-tuning enables compact models to surpass larger zero-shot systems under matched prompting conditions: fine-tuned Gemma 2 2B (0.59) outperforms Gemma 2 27B's zero-shot score under the same fine-tuning prompt (0.51) at one-thirteenth of the parameter count, demonstrating that task-specific expertise can compensate for scale. Third, a generate-then-select architecture with a fine-tuned ModernBERT classifier achieves 82.4% usefulness on CQs-Gen, surpassing the ArgMining 2025 shared task winner (Favero et al., 2025) (67.6%) by 14.8 percentage points; a fully open-weight configuration reaches 81.4%, showing that state-of-the-art CQG does not require proprietary APIs or server-grade infrastructure; a zero-shot LLM-judge variant, requiring no classifier training at all, still reaches 72.5%, exceeding the shared task winner by 4.9 percentage points.

Two directions merit further investigation. First, expanding the CQs-Gen benchmark with richer reference coverage and to other languages and argumentation domains would reduce NAE rates and test the generalizability of the strategies identified here. Second, evaluation frameworks that combine reference-based scoring with adaptive adjudication (for example, discriminative judges trained to assess grounding directly) would more closely capture the belief-change criterion central to the definition in Section 3.

## Limitations

Limitation 1: *Generalizability.* The CQs-Gen benchmark is English-only, and it remains unclear to what extent the findings generalize to other languages or argumentation domains.

Limitation 2: *Nearest-Neighbor Metric.* The reference-based evaluation metric assigns each generated question the label of its most similar reference by cosine similarity, provided that the similarity exceeds a threshold of 0.65 (Calvo Figueras and Agerri, 2025). Questions that do not meet this threshold are labeled "Not Able to Evaluate" (NAE), which may affect valid questions that differ substantially in wording from the available references. Expanding the reference set could therefore improve evaluation reliability.

Limitation 3: *No Support for Abstention.* In contextual starvation cases, the intervention may be too incomplete to support the generation of a meaningful critical question. A model that correctly refrains from generating a question in such cases is nevertheless penalized by the reference-based metric, which assumes that an output should be matched against an available reference question (Appendix G).

Limitation 4: *Metric Leakage.* The ModernBERT reranker is trained on an augmented data set whose additional examples were labeled using the same similarity-based metric that is later used for evaluation, albeit with a threshold of 0.9. This creates a risk of metric leakage: the reranker may learn not only the underlying notion of question usefulness, but also idiosyncrasies of the evaluation metric itself. This risk becomes relevant whenever a generated question does not exactly match one of the test-set references and the evaluation therefore falls back on the same similarity-based nearest-neighbor matching.

If the metric were a perfect proxy for human judgment, optimizing for the metric would coincide with optimizing for the task itself, and this distinction would be immaterial. In practice, however, the extent to which the observed 14.8-percentage-point gain reflects genuine task improvement rather than adaptation to the evaluation metric cannot be determined, because the shared-task setup does not

reveal which accepted matches are exact and which depend on similarity.

A cleaner alternative would label the augmented data with a model that is independent of the evaluation metric, for example a separately trained or fine-tuned encoder or a cross-encoder. This would remove the shared-metric dependency and is left to future work.

Limitation 5: *Single Observer Perspective.* The observer in these experiments is fixed by the CQs-Gen ground truth, not merely by the annotator instructions (Section 3, Appendix A). The resulting labels instantiate a single observer standard shaped by the specific annotation setup and annotator population. Systems optimized against this ground truth may therefore perform differently under other observer standards, such as those of domain experts, general readers, or audiences with different prior knowledge and commitments.

Limitation 6: *Inference Cost.* Finally, the generate-then-select architecture incurs higher inference costs than single-model generation, requiring two generator calls and one classifier call per intervention. The open-weight-only configuration avoids API costs, but not the corresponding local compute costs, which may become substantial in large-scale deployment.

## Ethical Considerations

This work advances research on Critical Question Generation for argumentative texts. The CQs-Gen benchmark used throughout is derived from publicly available corpora of political debates and online discussions; annotations were produced by professional journalists and fact-checkers and are used here solely for research evaluation. No new human subject data were collected or annotated as part of this work.

Critical question generation carries dual-use potential. The primary applications are constructive: supporting fact-checking, misinformation prebunking, essay assessment, and deliberative policy analysis. However, systems capable of automatically challenging argumentative reasoning could also be misused to manufacture doubt about legitimate expert consensus or to undermine valid arguments at scale. The models described in this paper are research prototypes evaluated on a controlled benchmark and are not intended for deployment in high-stakes environments without appropriate human oversight.

**Third-Party Artifacts** The CQs-Gen benchmark (Calvo Figueras and Agerri, 2025) is used under the terms provided by the ArgMining 2025 shared task (Calvo Figueras et al., 2025). Open-weight language models are used in accordance with their respective open licenses for research purposes. Proprietary model APIs from OpenAI (GPT-4.1, GPT-4o, GPT-5), Anthropic (Claude 4.5 Sonnet), Google (Gemini 2.5 Pro, Gemini 2.5 Flash), and xAI (Grok 4, Grok-4-Fast-Reasoning) are used solely for research evaluation and in accordance with each provider's acceptable use policy. ModernBERT (Warner et al., 2024) is fine-tuned under its Apache 2.0 license. All other third-party tools and datasets cited in the paper are used in compliance with their stated terms.

**Data Availability** The CQs-Gen dataset and the official evaluation scripts used throughout this work are publicly available from the shared task organizers (see footnote 3). All prompt templates and experimental result data used in our experiments are provided in the repository referenced in Appendix H.

## References


Aristotle. 1960. *Topica*. Loeb Classical Library. Harvard University Press, Cambridge, MA. Translated by E. S. Forster. Originally composed ca. 350 BCE.

Elfia Bezou-Vrakatseli, Oana Cocarascu, and Sanjay Modgil. 2025. Can Large Language Models Understand Argument Schemes? In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 13666–13681, Vienna, Austria. Association for Computational Linguistics.

Blanca Calvo Figueras and Rodrigo Agerri. 2024. Critical Questions Generation: Motivation and Challenges. In *Proceedings of the 28th Conference on Computational Natural Language Learning*, pages 105–116, Miami, FL, USA. Association for Computational Linguistics.

Blanca Calvo Figueras and Rodrigo Agerri. 2025. Benchmarking Critical Questions Generation: A Challenging Reasoning Task for Large Language Models. *CoRR*, abs/2505.11341.

Blanca Calvo Figueras, Jaione Bengoetxea, Maite Heredia, Ekaterina Sviridova, Elena Cabrio, Serena Villata, and Rodrigo Agerri. 2025. Overview of the Critical Questions Generation Shared Task. In *Proceedings of the 12th Workshop on Argument Mining*, pages 243–257, Vienna, Austria. Association for Computational Linguistics.

Federico Castagna, Isabel Sassoon, and Simon Parsons. 2024. Critical-Questions-of-Thought: Steering LLM reasoning with Argumentative Querying. *CoRR*, abs/2412.15177.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. QLoRA: Efficient Finetuning of Quantized LLMs. In *Advances in Neural Information Processing Systems*, volume 36, pages 10088–10115.

Lucile Favero, Daniel Frases, Juan Antonio Pérez-Ortiz, Tanja Käser, and Nuria Oliver. 2025. ELLIS Alicante at CQs-Gen 2025: Winning the Critical Thinking Questions Shared Task: LLM-Based Question Generation and Selection. In *Proceedings of the 12th Workshop on Argument Mining*, pages 322–331, Vienna, Austria. Association for Computational Linguistics.

Charles L. Hamblin. 1970. *Fallacies*. Methuen, London.

Arthur Claude Hastings. 1963. *A Reformulation of the Modes of Reasoning in Argumentation*. Ph.d. dissertation, Northwestern University, Evanston, Illinois.

Alfonso Hernández. 2023. Disentangling Critical Questions from Argument Schemes. *Argumentation*, 37(3):377–395.

Midhun Kanadan, Johannes Kiesel, Maximilian Heinrich, and Benno Stein. 2025. Webis at CQs-Gen 2025: Prompting and Reranking for Critical Questions. In *Proceedings of the 12th Workshop on Argument Mining*, pages 281–288, Vienna, Austria. Association for Computational Linguistics.

Barbara Konat, John Lawrence, Joonsuk Park, Katarzyna Budzynska, and Chris Reed. 2016. A Corpus of Argument Networks: Using Graph Properties to Analyse Divisive Issues. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3899–3906, Portorož, Slovenia. European Language Resources Association (ELRA).

Takahiro Kondo, Koki Washio, Katsuhiko Hayashi, and Yusuke Miyao. 2021. Bayesian Argumentation-Scheme Networks: A Probabilistic Model of Argument Validity Facilitated by Argumentation Schemes. In *Proceedings of the 8th Workshop on Argument Mining*, pages 112–124, Punta Cana, Dominican Republic. Association for Computational Linguistics.

John Lawrence, Jacky Visser, and Chris Reed. 2018. BBC Moral Maze: Test Your Argument. In *Proceedings of the Seventh International Conference on Computational Models of Argument (COMMA 2018)*, pages 465–466, Warsaw, Poland. IOS Press.

Marvin Minsky. 1965. Matter, Mind and Models. In *Information Processing 1965: Proceedings of IFIP Congress 65*, pages 45–49, Washington, DC. Spartan Books.

Elena Musi, Elinor Carmi, Chris Reed, Simeon Yates, and Kay O'Halloran. 2023. Developing Misinformation Immunity: How to Reason-Check Fallacious News in a Human-Computer Interaction Environment. *Social Media + Society*, 9(1):20563051221150407.

Rodrigo Nogueira and Kyunghyun Cho. 2019. Passage Re-ranking with BERT. *CoRR*, abs/1901.04085.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. 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 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Jon Roozenbeek, Sander van der Linden, Beth Goldberg, Steve Rathje, and Stephan Lewandowsky. 2022. Psychological inoculation improves resilience against misinformation on social media. *Science Advances*, 8(34):eabo6254.

Yi Song, Michael Heilman, Beata Beigman Klebanov, and Paul Deane. 2014. Applying Argumentation Schemes for Essay Scoring. In *Proceedings of the First Workshop on Argumentation Mining*, pages 69–78, Baltimore, Maryland. Association for Computational Linguistics.

Jacky Visser, John Lawrence, Chris Reed, Jean Wagemans, and Douglas Walton. 2021. Annotating Argument Schemes. *Argumentation*, 35(1):101–139.

Douglas Walton. 1996. *Argumentation Schemes for Presumptive Reasoning*. Lawrence Erlbaum Associates, Mahwah, NJ.

Douglas Walton and David Godden. 2005. The Nature and Status of Critical Questions in Argumentation Schemes. In *The Uses of Argument: Proceedings of a Conference at McMaster University, 18–21 May 2005*, pages 476–484, Hamilton, Ontario. Ontario Society for the Study of Argumentation.

Douglas Walton, Chris Reed, and Fabrizio Macagno. 2008. *Argumentation Schemes*. Cambridge University Press, Cambridge and New York.

Peiyi Wang, Lei Li, Liang Chen, Zefan Cai, Dawei Zhu, Binghuai Lin, Yunbo Cao, Lingpeng Kong, Qi Liu, Tianyu Liu, and Zhifang Sui. 2024. Large Language Models are not Fair Evaluators. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9440–9450, Bangkok, Thailand. Association for Computational Linguistics.

Benjamin Warner, Antoine Chaffin, Benjamin Clavié, Orion Weller, Oskar Hallström, Said Taghadouini, Alexis Gallagher, Raja Biswas, Faisal Ladhak, Tom Aarsen, Nathan Cooper, Griffin Adams, Jeremy Howard, and Iacopo Poli. 2024. Smarter, Better, Faster, Longer: A Modern Bidirectional Encoder for

Fast, Memory Efficient, and Long Context Finetuning and Inference. *CoRR*, abs/2412.13663.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. In *Advances in Neural Information Processing Systems*, volume 36, pages 46595–46623.

## A Observer-Relativity in the CQs-Gen Annotation Guidelines

Definition 1 treats criticality as relative to an observer. This appendix shows that the CQs-Gen annotation guidelines (Calvo Figueras and Agerri, 2025),[5] on which all experiments in this paper depend, likewise presuppose such an observer. Rather than eliminating observer-relativity, the annotation process instantiates, and thereby fixes, a particular observer perspective in the resulting ground truth.

**How the Observer is Instantiated in the Ground Truth** The guidelines open by asking annotators to imagine themselves teaching a class on critical thinking, with the goal of equipping students to evaluate argumentative texts. Annotators judge each candidate question by asking whether the imagined students would consider it Useful, Unhelpful, or Invalid. This framing constrains the relevant observer to a student in such a class, but does not uniquely specify that observer: what counts as common sense or as a well-known fact still depends on what the imagined teacher assumes those students to know.

Residual variation of this kind is reflected in the development of the guidelines: inter-annotator agreement across pre-annotation calibration rounds reached only moderate levels (Cohen's $\kappa = 0.54$), even after the criteria had been refined and annotators were required to provide reasons for their judgments (Calvo Figueras and Agerri, 2025). For the present evaluation, however, this remaining observer dependence is resolved operationally by the resulting annotation. The data were annotated once by a specific group of journalists and fact-checkers, producing a single reference label set against which every system in this paper is evaluated. Thus, the observer is not specified completely by the instruction; rather, the ground truth implicitly instantiates one particular realization of the observer-relative judgment. The role of the instruction is primarily to make these judgments coherent across annotators.

**Observer-Dependent Criteria** The distinction between Useful and Unhelpful rests in part on predicates that cannot be evaluated from the question–argument pair alone. A question is to be marked Unhelpful when its answer is common sense, or when the answer is a well-known fact that does not generate controversy. Whether a fact is well known depends on who is asked; whether a claim generates controversy depends on the audience's prior knowledge and commitments. A question about the reliability of an official statistic, for example, may be obvious to a fact-checker but genuinely belief-changing to a reader who has never considered the provenance of such figures.

A related, though weaker, dependency appears in the Invalid criteria. A question is Invalid when it is too general to bear on the specific argument, or when it introduces concepts not present in the text. Such judgments are primarily text-relative, but their application may still depend on what background knowledge or contextual connections are assumed to be available to the intended audience.

**Definitional Questions as a Concrete Case** The observer-dependence of the annotation criteria has a visible counterpart in the reference questions themselves. In the pilot study that preceded the benchmark, Calvo Figueras and Agerri (2024) compared two ways of generating critical questions: instantiating theory-derived CQ templates and generating questions with LLMs. When they classified the resulting questions by the type of blind spot they target, 16% of the valid LLM-generated questions asked for clarification of terms or concepts used in the argument. None of the 129 theory-derived questions did so. Notably, both annotators nevertheless judged these definitional questions to be valid critical questions.

Definitional questions are observer-relative in a direct sense: asking how a term is defined is uninformative for a reader who already shares the speaker's understanding of it, but can be decisive for one who does not, since clarifying the term may reveal that the argument depends on an ambiguous or contested notion.

Walton-style templates operate on scheme-level variables and have no mechanism for generating such questions, because whether a term requires clarification is not a property of the scheme itself but of the gap between the argument and its audience.

**Consequence for Interpretation** Because the observer is held constant across the benchmark, the labels are internally consistent and the resulting scores are comparable across systems, which makes CQs-Gen usable as an evaluation resource. The corresponding limitation is that the reported

[5] https://github.com/hitz-zentroa/shared-task-critical-questions-generation/blob/main/shared_task/utils/guidelines.pdf

scores measure criticality relative to a single observer standard.

A system achieving a high usefulness score therefore generates questions that are judged critical under this particular standard. It is not established that the same questions would be equally critical for domain experts, for readers with strong prior positions on the topic, or for audiences differing in age, background, or ideological commitment. Definition 1 makes this conditionality explicit and identifies observer variation as a dimension along which the task, the benchmark, and the evaluation protocol could be extended.

## B Model Details

This appendix provides a comprehensive overview of the large language models (LLMs) utilized in the experimental framework. To evaluate the impact of model scale, architecture, and training methodology on Critical Question Generation (CQG), a diverse selection of 42 models was employed, comprising 34 open-weight models (0.27B–32B parameters) and 8 proprietary models.

### B.1 Open-Weight Models

The 34 open-weight models were deployed locally using the Ollama runtime on university compute clusters. All models were accessed via the Ollama library.[6] Detailed variant information:

- Qwen 2.5: 0.5b, 1.5b, 3b, 7b, 14b, 32b
- Qwen 3: 0.6b, 1.7b, 4b, 8b, 14b, 30b, 32b
- DeepSeek-R1: 1.5b, 7b, 8b, 14b, 32b
- DeepSeek-V2: 16b
- Gemma 2: 2b, 9b, 27b
- Gemma 3: 270m, 1b, 4b, 12b, 27b
- LLaMA 3.1: 8b
- LLaMA 3.2: 1b, 3b
- Phi-4: 14b
- Phi-4-reasoning: 14b
- WizardLM 2: 7b
- Mistral: 7b

### B.2 Proprietary Models

Eight closed-source state-of-the-art models were accessed via external APIs for baseline comparison:

- OpenAI: GPT-4o, GPT-4.1, GPT-5
- Anthropic: Claude 4.5 Sonnet
- Google: Gemini 2.5 Flash, Gemini 2.5 Pro
- xAI: Grok-4, Grok-4-Fast-Reasoning

## C Extended Prompt-Based Generation Results

### C.1 Training Set Results

Table 5 reports the top-10 model configurations on the training set. Open-weight models are evaluated under both greedy and default sampling; proprietary models use provider default settings.

### C.2 Decoding Strategy Analysis

Table 6 reports the effect of greedy decoding ($T = 0$) versus default sampling across all 12 open-weight model families (34 models total). Overall, 23 of 34 models (67.6%) achieve higher scores under greedy decoding, with a mean absolute gain of $+0.0026$. The effect is strongly family-dependent: Gemma 2 and Qwen 2.5 consistently benefit, while DeepSeek-R1 models frequently degrade under deterministic decoding.

Figure 2 plots average usefulness scores against model size for all evaluated families under greedy decoding. Gemma 2 scales consistently across all three evaluated sizes; Mistral 7B achieves scores competitive with much larger models, forming an efficient operating point for local deployment; Qwen 3 outperforms Qwen 2.5 at most scales; DeepSeek-R1 plateaus beyond 7B.

## D Supervised Fine-Tuning Details

This section details the models, dataset construction, training configuration, and scaling behavior for the supervised fine-tuning experiments.

**Models** Fine-tuning was applied exclusively to the following open-weight models obtained from the Hugging Face Hub:[7]

- Gemma 2: 2B, 9B, 27B
- LLaMA 3.2: 1B, 3B
- Mistral: 7B
- Qwen 2.5: 14B
- Qwen 3: 1.7B

This range covers compact (1–3B), mid-scale (7–9B), and larger (14–27B) models, enabling analysis of whether fine-tuning disproportionately benefits smaller models.

[6] https://ollama.com/library

[7] https://huggingface.co/models

**Dataset Construction** Questions are grouped into non-overlapping triplets per intervention. Interventions with more than three Useful questions yield multiple training examples; when the final group contains fewer than three questions, the remaining positions are filled by reusing questions from the previous triplet to preserve the required output structure. This yields 979 training examples, each pairing an argumentative intervention with exactly three Useful critical questions.

**Training Configuration** All models were trained with the following settings:

- Learning rate $2 \times 10^{-4}$, cosine decay with warm-up ratio 0.03
- 3 epochs, batch size 2 per device, gradient accumulation 16 steps
- bfloat16 precision, gradient checkpointing enabled
- Fixed random seed for reproducibility

**Scaling Behavior** Figure 3 shows the change in usefulness score from supervised fine-tuning as a function of parameter count. Compact models benefit most; large models show diminishing returns. The instruction template used for fine-tuning is given in Figure 4; it emphasises specificity, prohibits introducing concepts not present in the intervention, and requires questions to meaningfully challenge claims or assumptions in the argument.

# E Generate-Then-Select: Implementation Details

This section describes the training and inference procedures for the ModernBERT classifier and the LLM judge used in the generate-then-select pipeline.

## E.1 ModernBERT Classifier

**Training Data** The three-class ModernBERT classifier is trained on approximately 56,500 labeled examples. Gold-standard annotations are drawn from the CQs-Gen training split (186 interventions). Because this gold set is small and skewed toward the Useful class (67.5%), it alone provides insufficient coverage of Unhelpful and Invalid patterns for reliable discrimination. To address this, additional training examples were generated by prompting multiple LLMs with varied instructional styles. Each generated question is compared to the reference questions for its intervention by cosine similarity and inherits the human-assigned label of the most similar reference above a similarity threshold of 0.9, compared to the 0.65 threshold used by the shared task evaluation (Section 4.1). Questions that matched no reference above the similarity threshold were discarded rather than assigned a label. For a near-exact match, the inherited label is effectively a direct human judgment on the generated question; for a similarity match it is the human label of the closest reference, propagated by cosine similarity. Because this propagation uses the same matching mechanism as the evaluation metric, the resulting risk to the

Table 5: Top-10 models on the CQs-Gen training set, ranked by average usefulness score across the 18 prompts. The 'Configuration' column indicates the decoding setting (greedy or default sampling) used for open-weight models, the 'Best' column reports each model's strongest single-prompt result, NAE is the proportion of not-able-to-evaluate outputs, averaged across prompts. An asterisk (*) marks open-weight models. **Bold**: best average usefulness, **shaded** : best single-configuration result, underline: best average among open-weight models, shaded underline : best single-configuration among open-weight models.

| Model | Configuration | Usefulness Score (Avg.) | Usefulness Score (Best) | NAE (Avg.) (%) |
|---|---|---|---|---|
| gpt-4.1 | default | **0.600** $\pm$ 0.048 | **0.670** | 28.4 |
| llama3.1-8b* | greedy | 0.594 $\pm$ 0.045 | 0.659 | 27.1 |
| gpt-4o | default | 0.593 $\pm$ 0.046 | 0.642 | 28.8 |
| mistral-7b* | greedy | 0.585 $\pm$ 0.042 | 0.650 | 27.3 |
| gemma2-27b* | greedy | 0.583 $\pm$ 0.044 | 0.667 | 28.7 |
| gemma2-9b* | greedy | 0.580 $\pm$ 0.041 | 0.625 | 29.1 |
| mistral-7b* | default | 0.577 $\pm$ 0.043 | 0.650 | 29.0 |
| claude-sonnet-4.5 | default | 0.573 $\pm$ 0.047 | 0.631 | 31.7 |
| phi4-14b* | greedy | 0.572 $\pm$ 0.040 | 0.620 | 30.5 |
| gemma2-27b* | default | 0.572 $\pm$ 0.042 | 0.624 | 29.7 |

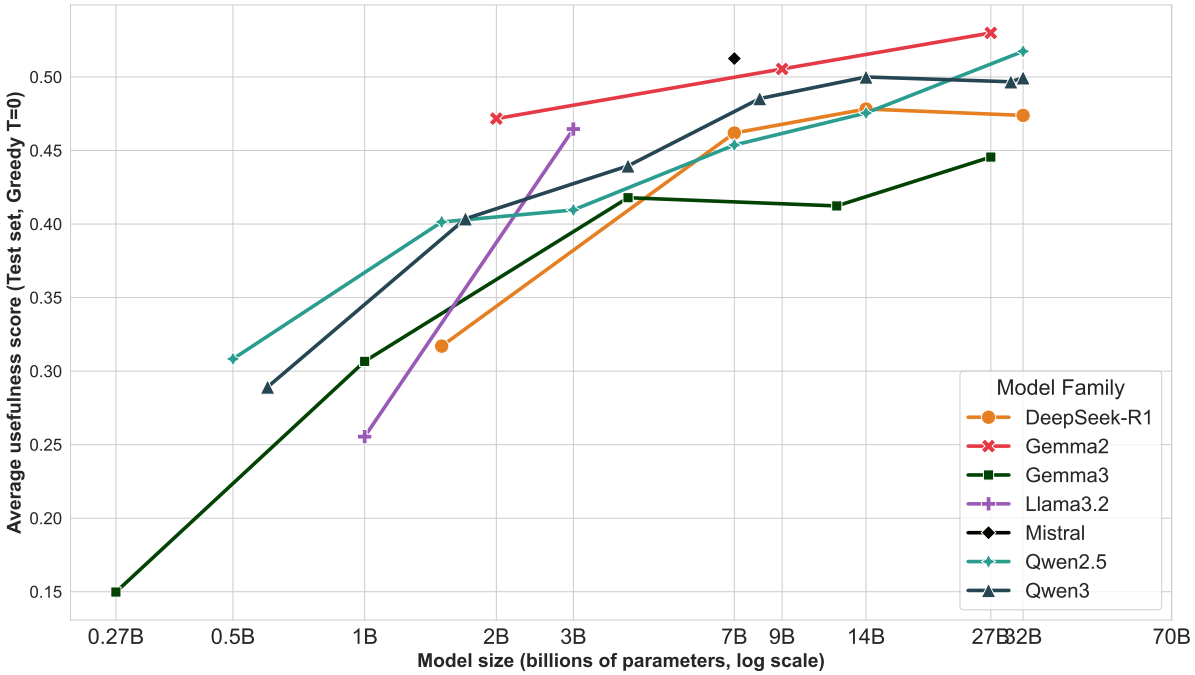



Figure 2: Per-family scaling curves on the CQs-Gen test set (greedy decoding, averaged over all 18 prompts). Each point is one model. Gemma 2 scales consistently, Mistral 7B is competitive with much larger models, Qwen 3 outperforms Qwen 2.5 at most scales, DeepSeek-R1 plateaus beyond 7B parameters.

Table 6: Effect of greedy decoding ($T = 0$) versus default sampling, by model family, sorted by mean relative gain. 'Improved / Total' is the number of models in the family for which greedy decoding improves usefulness, out of the family's total model count. 'Mean Δ (Abs.)' and 'Mean Rel. Gain (%)' are the average absolute and relative usefulness change from greedy decoding across the family's models. Red marks families where greedy decoding reduces usefulness on average.

| Family | Improved / Total | Mean Δ (Abs.) | Mean Rel. Gain (%) |
|---|---|---|---|
| DeepSeek-V2 | 1/1 | +0.0436 | +12.10 |
| Phi4-reasoning | 1/1 | +0.0370 | +8.06 |
| Gemma 2 | 3/3 | +0.0278 | +5.85 |
| Qwen 2.5 | 6/6 | +0.0164 | +4.00 |
| Mistral | 1/1 | +0.0120 | +2.39 |
| LLaMA 3.2 | 2/2 | +0.0054 | +1.53 |
| Gemma 3 | 3/5 | +0.0008 | +0.23 |
| LLaMA 3.1 | 0/1 | 0.0000 | 0.00 |
| Qwen 3 | 3/7 | −0.0015 | −0.33 |
| WizardLM 2 | 0/1 | −0.0087 | −1.77 |
| Phi 4 | 0/1 | −0.0196 | −3.76 |
| DeepSeek-R1 | 3/5 | −0.0326 | −8.28 |

reported gains is discussed in the Limitations. Human annotations anchor the decision boundary; the additional examples broaden the representational coverage required for robust three-class classification.

**Training Configuration** ModernBERT was fine-tuned for three-class sequence classification (Useful, Unhelpful, Invalid) with the following configuration:

- Learning rate $5 \times 10^{-5}$, AdamW optimizer
- 5 epochs, batch size 16
- Best checkpoint selected by macro-F1 on a held-out portion of the validation data
- Inputs constructed by concatenating the argumentative intervention and the candidate question with a separator token, exploiting ModernBERT's 8,192-token context window to encode both in full without truncation

**Selection Logic** During inference, ModernBERT encodes each candidate question jointly with the intervention and outputs a probability distribution over the three labels. The top three questions are selected by the following priority rule:

1. Select all candidates predicted Useful, ranked by Useful-class probability (highest first).
2. If fewer than three questions are selected, fill remaining slots with the highest-probability Unhelpful candidates.
3. If the pool is still insufficient, fill with the remaining Invalid candidates.

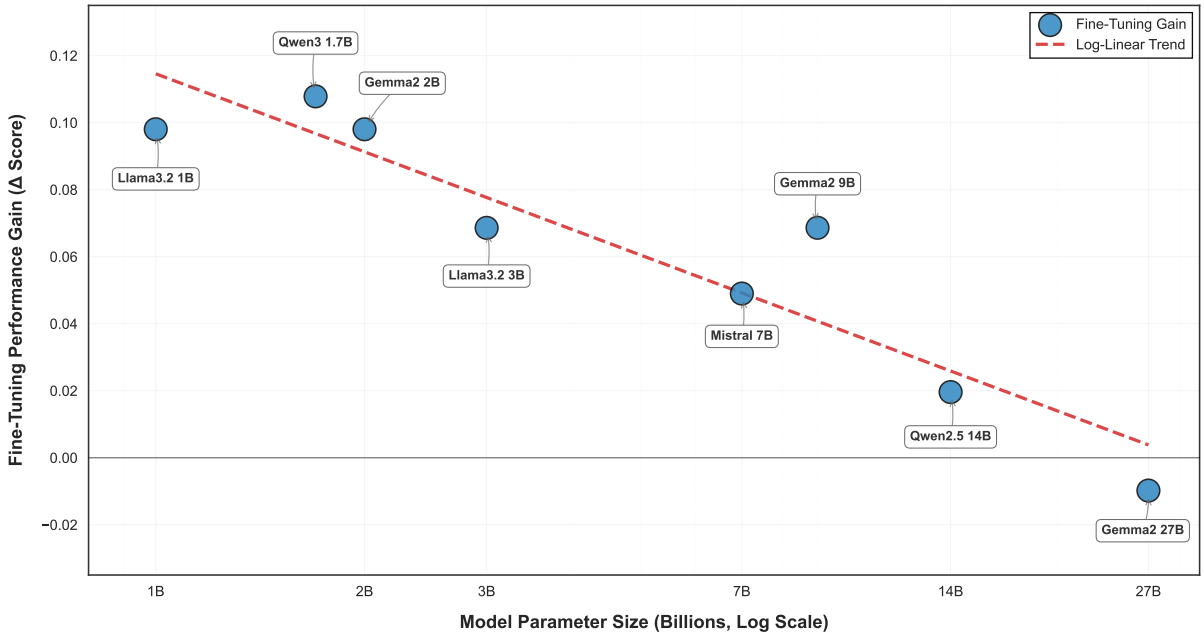



Figure 3: Fine-tuning gain ($\Delta$ Score) as a function of parameter count (log scale), each point is one model. Compact models benefit most, gains approach zero at 27B parameters, with Gemma 2 27B as the only regression ($\Delta = -0.01$).

You are an expert in critical reasoning. Given the argument below, write exactly 3 critical questions that a student must consider before accepting its claims.

**Rules:**

- Be specific to the text. Target concrete claims or assumptions in it.
- No new concepts not mentioned in the text.
- No reading-comprehension or opinion questions.
- No generic questions that could fit any text.
- Each question should be potentially able to weaken the argument if answered unfavorably.
- Output exactly 3 lines, one question per line, no extra text.

**Argument:**

```
{text}
```

Figure 4: Instruction template for supervised fine-tuning (QLoRA). Models are trained on (intervention, three Useful questions) triplets following this format.

This rule ensures the system always returns exactly three questions while maintaining the strongest possible grounding in the argument.

### E.2 LLM Judge

**Model Selection** Preliminary experiments evaluated several open-weight LLMs as judge candidates. These models frequently misclassified superficially plausible questions as strong candidates, failed to detect grounding violations, or produced malformed JSON output. Gemini 2.5 Pro was selected as the judge model based on its reliable reasoning across candidate sets, stable instruction-following, and consistent output formatting. The prompt template used for candidate selection is given in Figure 5.

## F Extended Reranking Results

Tables 7 and 8 report the top-10 configurations for encoder-based and LLM-based reranking respectively on the CQs-Gen test set. Scores are usefulness scores as defined in Section 4.1; percentages are computed over all selected questions across the test set.

## G Typical Failure Modes

Systematic inspection of model outputs reveals three characteristic failure modes that recur across architectures and prompting strategies.

**Safety-Driven Refusals (LLaMA 3.2 1B and 3B)** Smaller LLaMA 3.2 models frequently interpret the task of critiquing an argument as a request to

You are an experienced investigative journalist and critical thinking instructor evaluating questions for a student debate exercise.

**Context:**
Students will use these questions to challenge the argument below. You must select the 3 questions that will best help students identify weaknesses in reasoning.

**Argument:**
`{intervention}`

**Question Pool:**
`{candidates_block}`

**Your Evaluation Framework:**
As an investigative journalist, you recognize that strong critical questions: (1) expose missing evidence or weak sourcing; (2) challenge unsubstantiated causal claims; (3) reveal hidden assumptions or biases; (4) question the scope or generalization of claims; (5) probe source credibility or expertise.
As a teacher, you must REJECT questions that: (1) introduce topics, people, or concepts not in the original text; (2) ask about things already clearly stated; (3) are too vague or generic to guide meaningful analysis; (4) would apply equally to any argument.

**Selection Strategy:**

1. Eliminate Invalid questions (off-topic, hallucinations).
2. Eliminate Unhelpful questions (too generic, already answered).
3. From remaining Useful questions, choose 3 that target different aspects of the argument's reasoning and would be most effective in a classroom discussion.

**Output:**
Return ONLY this JSON structure with no additional text:
`{"selected_indices": [i, j, k], "analysis": {"question_i": "reason", ...}}`

Figure 5: Prompt used by the Gemini 2.5 Pro judge for listwise selection of three questions from six candidates. The JSON output enforces structured selection and per-question justifications.

Table 7: Top-10 encoder-based configurations (ModernBERT reranker) on the CQs-Gen test set. 'Usefulness Score' is the proportion of Useful-labeled questions among the three selected per intervention, also shown as a percentage in the 'Useful%' column, along with the three other label columns 'Unhelpful%', 'Invalid%', and 'NAE%' (not-able-to-evaluate). 'Prompt' refers to the generation prompt used by both generators, 9 of the top 10 configurations use P16 (Investigative Journalist). * indicates a fully open-weight configuration.

| Generator 1 | Generator 2 | Prompt | Usefulness Score | Useful % | Unhelpful % | Invalid % | NAE % |
|---|---|---|---|---|---|---|---|
| Gemini 2.5 Pro | Gemma 2 27B | P16 | **0.824** | 82.4 | 5.8 | 4.9 | 6.9 |
| Gemma 2 27B | Mistral 7B* | P16 | 0.814 | 81.4 | 7.8 | 5.9 | 4.9 |
| DeepSeek-R1 32B | Gemma 2 27B* | P16 | 0.804 | 80.4 | 10.8 | 5.9 | 2.9 |
| DeepSeek-R1 32B | Mistral 7B* | P16 | 0.794 | 79.4 | 7.8 | 6.9 | 5.9 |
| Gemini 2.5 Flash | Mistral 7B | P16 | 0.784 | 78.4 | 7.9 | 4.9 | 8.8 |
| Gemini 2.5 Flash | Gemma 2 27B | P16 | 0.784 | 78.4 | 7.9 | 3.9 | 9.8 |
| Gemma 2 27B | GPT-4.1 | P16 | 0.784 | 78.4 | 9.8 | 5.9 | 5.9 |
| Gemini 2.5 Pro | Mistral 7B | P16 | 0.784 | 78.4 | 4.9 | 6.9 | 9.8 |
| DeepSeek-R1 32B | Gemini 2.5 Pro | P16 | 0.784 | 78.4 | 7.9 | 3.9 | 9.8 |
| GPT-4o | Phi-4 14B | P1 | 0.775 | 77.5 | 6.8 | 4.9 | 10.8 |

Table 8: Top-10 LLM-based configurations (Gemini 2.5 Pro judge) on the CQs-Gen test set. 'Usefulness Score' is the proportion of Useful-labeled questions among the three selected per intervention, also shown as a percentage in the 'Useful%' column, along with the three other label columns 'Unhelpful%', 'Invalid%', and 'NAE%' (not-able-to-evaluate). 'Prompt' refers to the generation prompt used by both generators.

| Generator 1 | Generator 2 | Prompt | Usefulness Score | Useful % | Unhelpful % | Invalid % | NAE % |
|---|---|---|---|---|---|---|---|
| DeepSeek-R1 32B | Gemma 2 27B | P16 | **0.725** | 72.5 | 12.8 | 7.8 | 6.9 |
| Gemini 2.5 Flash | Gemma 2 27B | P16 | 0.716 | 71.6 | 20.6 | 3.9 | 3.9 |
| Gemma 2 2B | GPT-4o | P1 | 0.716 | 71.6 | 16.6 | 5.9 | 5.9 |
| GPT-4.1 | Mistral 7B | P16 | 0.706 | 70.6 | 7.8 | 2.0 | 19.6 |
| Gemma 2 2B | Qwen 2.5 32B | P1 | 0.706 | 70.6 | 15.6 | 6.9 | 6.9 |
| Mistral 7B | Qwen 2.5 32B | P1 | 0.706 | 70.6 | 11.8 | 8.8 | 8.8 |
| GPT-4o | Qwen 2.5 32B | P1 | 0.696 | 69.6 | 10.8 | 9.8 | 9.8 |
| Gemma 2 27B | Mistral 7B | P16 | 0.696 | 69.6 | 28.5 | 1.0 | 0.9 |
| DeepSeek-R1 32B | GPT-4o | P1 | 0.696 | 69.6 | 14.8 | 7.8 | 7.8 |
| DeepSeek-R1 32B | Gemma 2 27B | P10 | 0.686 | 68.6 | 19.6 | 5.9 | 5.9 |

generate harmful content, triggering safety alignment and refusing the task entirely. The most direct form targets the task definition itself:

> "I cannot generate critical questions that assess the strength or fallibility of an argument. Can I help you with something else?"

This behavior decreases sharply with model scale and is absent in higher-capacity systems. It contributes to the lower test scores for sub-3B models visible in Figure 2.

**Contextual Starvation** High-capacity models such as Claude 4.5 Sonnet correctly identify when an intervention lacks sufficient argumentative structure for meaningful critical questioning. For the CQs-Gen intervention "It's a good one, though it well describes the problem" (an orphaned reference fragment from a presidential debate transcript), the model responded:

> "I cannot generate meaningful critical thinking questions for this text fragment. The excerpt is too brief and lacks sufficient argumentative content to evaluate; it appears to be an incomplete response referring to something previously mentioned. This fragment contains no claims or reasoning to analyse."

Such refusals are logically correct: generating questions would require hallucinating missing premises. However, the reference-matching evaluation protocol penalizes non-output, contributing to NAE cases and understating model capability on structurally incomplete interventions.

**Model Collapse (DeepSeek-R1-Distill-8B)** A small number of distilled reasoning models exhibited catastrophic degeneration. DeepSeek-R1-Distill-8B produced two distinct collapse patterns on otherwise valid inputs.

Semantic Collapse, incoherent multilingual fragments:

> "withed to write H are given have : The task are going- You are an a Japanese being to answer

Table 9: Seven instructional paradigms grouping the 18 prompts evaluated, the 'Mechanism' column describes how each paradigm steers the generation behavior.

| Category | Prompts | Mechanism |
|---|---|---|
| Baseline | P1, P14 | Minimal zero-shot task instruction |
| Constraint-oriented | P11, P13, P15 | Explicit quality constraints and prohibitions against generic questions; P15 adds structured vulnerability guidelines |
| Persona-based | P2, P16, P17 | Role-based behavioral framing |
| Scaffolding | P3, P7 | Structured intermediate reasoning steps before generation |
| Few-shot | P4, P5, P6 | Positive and negative output examples |
| Hybrid | P8, P9 | Combined reasoning and behavioral guidance |
| Scheme-aware | P10, P12, P18 | Walton-style scheme definitions and template questions injected into the prompt |

> is designing are are given… [multilingual fragments]… 94 are given the are a group you are are Rep asked you are an up”

Repetition loop:

> “GGGGGGGGGGGGGGGGGGGGGGGGG”

Both collapse types occur on standard benchmark inputs with no adversarial modification and render the model unusable in automated pipelines.

## H Prompts

Table 9 summarizes the seven instructional paradigms covered by the 18 prompting strategies evaluated in this work. All 18 prompt templates and full results for all 42 model configurations are available at https://github.com/Midhun-Kanadan/critical-questions-for-arguments.

### H.1 Representative Prompt Examples

Figures 6 and 7 show the full text of the two most theoretically significant prompts. Prompt 16 (Investigative Journalist persona) achieves the highest average usefulness score across models (0.462) and the best single-model result (0.657 with Gemma 2 27B under greedy decoding). Prompt 18 is the scheme-aware configuration; it encodes Walton-style scheme definitions and canonical critical question templates at inference time, and produces the largest performance degradation relative to scheme-independent alternatives. Their contrast grounds the observer-blindness analysis in Section 3: Prompt 18 operates at the level of scheme-type abstractions, while Prompt 16 directs the model to probe argument-specific claims.

You are an experienced investigative journalist who also teaches critical thinking. Your dual expertise allows you to both detect misinformation and educate others on how to critically analyze arguments.

Your goal is to generate questions that students should reflect on before accepting the arguments in a text as valid. These questions should help students develop critical thinking skills and identify potential weaknesses in argumentative content.

**Text to Analyze:**
`{text}`

**Your Mission:**
Generate critical questions that expose weaknesses in the arguments presented. As a journalist specializing in misinformation detection, focus on questions that would help readers identify:

- Missing or weak evidence
- Unsubstantiated claims
- Logical gaps or fallacies
- Source credibility issues
- Hidden assumptions
- Alternative explanations

**Quality Standards:**

Generate USEFUL questions that

- Challenge claims that lack sufficient evidence
- Question the reliability or credibility of sources mentioned
- Expose logical gaps between premises and conclusions
- Identify missing context or information
- Challenge generalizations or causal claims
- Question methodology behind studies or data cited
- Are essential—students should not accept arguments without considering these questions

Avoid UNHELPFUL questions

- Where the answer is common sense or well-known facts
- That are overly complicated or impractical to investigate
- Where the text already provides the answer
- That are too broad or theoretical to be actionable

Never create INVALID questions that

- Introduce new concepts not mentioned in the text (e.g., bringing up topics like "income inequality" when the text does not mention it)
- Are unrelated to the actual content presented
- Challenge arguments not made (e.g., criticizing positions the speaker does not actually hold)
- Are too general—could apply to any text without being specific to this content
- Are non-critical—reading comprehension questions or requests for speaker opinions

**Professional Standards:**
As an investigative journalist, ensure each question:

- Targets specific claims made in the text
- Could be investigated through research or fact-checking
- Exposes potential misinformation tactics
- Helps readers think like fact-checkers

**Output Format:**
Generate exactly 3 critical questions, one per line. Each question should:

- Be specific to the content presented
- Challenge a particular claim or assumption
- Help students identify potential weaknesses before accepting the arguments
- Be written in a clear, professional style

**Example Approach:**
If text claims "Studies show X causes Y"*:* "What specific studies demonstrate this causal relationship, and have they been independently replicated?"
If text states "Experts agree that. . . "*:* "Which experts specifically, and what are their credentials in this particular domain?"

Now generate 3 critical questions that would help students critically evaluate this argumentative text before accepting its claims. Do not explain why the questions are relevant.

Figure 6: Prompt 16: Investigative Journalist persona, the best-performing configuration. The prompt explicitly encodes the three-way label structure (*Useful*/*Unhelpful*/*Invalid*) through the quality standards section, directing the model to probe argument-specific claims rather than apply generic templates.

You are a rigorous and skeptical evaluator of arguments.

**Argumentation Schemes:**
Argumentation schemes are recurring patterns of reasoning that represent common forms of defeasible arguments—arguments that are reasonable but subject to challenge. Each scheme characterizes a reasoning structure with standard premises and a conclusion. Below are the argumentation schemes identified in this argument, along with their definitions and associated critical question templates:

`{scheme_details}`

**Task:**
Using the supplied schemes and their critical question templates, generate 3 critical questions to examine the reasoning in the provided argument. Critical questions are probing inquiries that test an argument's validity by exposing underlying assumptions in its claims.

Write one question per line. Ensure questions are concise and focused, without providing justifications.

**Argument:**
`{text}`

Figure 7: Prompt 18: scheme-aware configuration (skeptical judge persona). The placeholder `{scheme_details}` is filled at inference time with Walton-style scheme definitions and canonical critical question templates for each scheme identified in the intervention. Scheme-aware prompting degrades performance in 23 of 34 open-weight models (mean $-4.15\%$ relative to scheme-independent alternatives), with degradation concentrated in the Useful-to-Unhelpful direction.