Webis at CQs-Gen 2025: Prompting and Reranking for Critical Questions

Midhun Kanadan[1] **Johannes Kiesel**[2] **Maximilian Heinrich**[1] **Benno Stein**[1]

[1]Bauhaus-Universität Weimar, [2]GESIS - Leibniz Institute for the Social Sciences

Abstract

This paper reports on the submission of team *Webis* to the Critical Question Generation shared task at the 12th Workshop on Argument Mining (ArgMining 2025). Our approach is a fully automated two-stage pipeline that first prompts a large language model (LLM) to generate candidate critical questions for a given argumentative intervention, and then reranks the generated questions as per a classifier's confidence in their usefulness. For the generation stage, we tested zero-shot, few-shot, and chain-of-thought prompting strategies. For the reranking stage, we used a ModernBERT classifier that we fine-tuned on either the validation set or an augmented version. Among our submissions, the best-performing configuration achieved a test score of 0.57 and ranked 5th in the shared task. Submissions that use reranking consistently outperformed baseline submissions without reranking across all metrics. Our results demonstrate that combining open-weight LLMs with reranking significantly improves the quality of the resulting critical questions.

1 Introduction

Large Language Models have demonstrated remarkable fluency in generating natural language text, but often struggle with hallucinations, outdated knowledge, or superficial reasoning (McKenna et al., 2023; Li et al., 2023; Islam et al., 2024). Therefore, one can not rely on LLMs to produce factual counterarguments. However, Critical Question Generation offers a different approach for arguing against statements: generating questions that expose an argument's "blind spots"—such as hidden assumptions, missing evidence, or flawed logic—which do not require factual knowledge to ask. Critical questions are thus not counterarguments in the typical sense of statements that are incompatible with the attacked argument. Instead, they are challenges to an argument's reasoning (Walton et al., 2008; Reed and Walton, 2001; Calvo Figueras and Agerri, 2024).

The ArgMining 2025 Shared Task on Critical Question Generation (Figueras et al., 2025) introduced a benchmark for evaluating automated question generation systems. Given interventions (contributions) to a debate, each annotated with argumentation schemes, submissions are required to generate three critical questions per intervention that meaningfully challenge the argument.

In this paper, we present our participating system (team *Webis*), which implements a two-stage pipeline: (1) prompting for critical questions and (2) reranking the generated questions to pick the most useful ones. For prompting, we test strategies ranging from basic zero-shot prompts to few-shot and chain-of-thought templates against multiple open-weight and closed-source models. For reranking, we use a ModernBERT classifier trained to predict usefulness on the shared task's validation dataset, as well as on an augmented version of this dataset of questions we generated and evaluated automatically.

Our system achieved 5th place in the shared task with an official score of 0.569, demonstrating the effectiveness of our two-stage pipeline.

This paper is structured as follows. Section 2 reviews related work on critical question generation and argumentation mining. Section 3 outlines the task definition. Section 4 presents our two-stage pipeline, detailing the prompting strategies and reranking. Section 5 reports our results, showing that reranking—especially when using an augmented training dataset—improved the effectiveness of methods over a baseline without reranking.

2 Related Work

The task of critical question generation is closely related to the notion of argumentation schemes (Walton et al., 2008), which define reasoning patterns

and associated critical questions that probe assumptions and implications. While critical questions have been explored in logic and pedagogy (Reed and Walton, 2001; Macagno et al., 2017), their automatic generation remains underexplored. Recent work by Calvo Figueras and Agerri (2024) introduced a shared task on critical question generation, comparing LLM-based generation with template-based instantiation of critical questions. Their study highlights the challenge of producing valid, relevant questions that challenge the logic of an argument.

A significant strand of work in computational argumentation has examined the detection of argumentative components such as claims, premises, and discourse relations (Lawrence and Reed, 2019). However, less attention has been paid to the generation of inferentially challenging questions. While datasets like US2016 and Moral Maze offer valuable annotations for argument structure and schemes (Visser et al., 2021), their limited size and coverage pose challenges for training robust models for critical question generation.

A typical choice of model for argument classification and evaluation tasks is BERT and its variants (Devlin et al., 2019), which have been widely used for stance classification, argument quality prediction, and claim detection. For instance, Favreau et al. (2022) utilized BERT-based learning-to-rank models to evaluate the convincingness of arguments, demonstrating its efficacy in ranking tasks. In our work, we extend this line of research by adapting a fine-tuned ModernBERT classifier to score the usefulness of generated critical questions.

In previous work, the shared task organizers (Calvo Figueras and Agerri, 2024) showed that LLMs can be used for critical question generation, but their outputs often lack inferential validity or relevance. Combining LLM-based generation with downstream filtering or reranking, as explored in this paper, has shown potential for improving quality and consistency (Jain et al., 2024).

3 Task

Given an argumentative text, the task of critical question generation is to generate three questions that directly challenge the argument. These texts are interventions from real-world debates.

To evaluate systems for critical question generation, each generated question for a given argument is matched to a set of reference questions—which were labeled as `Useful`, `Unhelpful`, or `Invalid`—using semantic similarity. Each generated question is assigned the label of its most similar reference question. A system's final score is computed as the proportion of generated questions labeled as `Useful` across all interventions. In addition to the annotated reference questions, the shared task's dataset also contains argument scheme annotations for each argument, which we used in one prompting strategy.

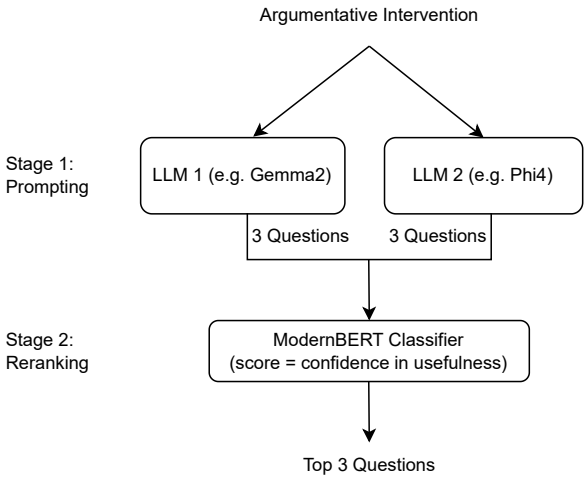

Figure 1: Overview of our evaluation and reranking pipeline. Each intervention is processed by two LLMs, generating six candidate critical questions. These are classified by a fine-tuned ModernBERT model, and the top three useful questions are selected.

4 Our Approach

Our approach is a pipeline of two stages: (1) prompting for critical questions and (2) reranking the generated questions to select the most useful ones. Figure 1 illustrates our pipeline.

4.1 Stage 1: Prompting for Critical Questions

To improve the quality, diversity, and relevance of generated critical questions, we implemented a wide spectrum of prompting strategies,[1] grouped into the following categories:

- *Basic prompting*: Directly asking the LLM to generate critical questions for an argumentative paragraph in a zero-shot manner.

[1]See Appendix B. As prompts are rather long, we only included the prompting strategy that led to the best results. The full set of prompts and code used in our system is available at `https://github.com/webis-de/argmining25-CQs-Gen.git`.

- *Guideline-based prompting*: Incorporating definitions of critical questions and their intended function (also zero-shot).

- *Chain-of-thought prompting*: Prompting the model to reason step-by-step before generating each question, for example by first identifying assumptions or implications in the argument (also zero-shot).

- *Few-shot prompting*: Providing one or more intervention–CQ examples, definitions with illustrative cases, good vs. bad comparisons, and self-assessment checks to guide the model toward higher-quality output.

These strategies were tested across different models and served as building blocks for the final prompt, which combined elements from several strategies—such as few-shot examples, definitions, and self-assessment instructions—to generate critical questions that are coherent, relevant, and inferentially valid.

Model selection: We used both open-weight and closed-source LLMs. The employed open-weight models—Gemma 3 (4B parameters), Gemma 2 (9B), LLaMA 3.2 (3B), Mistral (7B), Phi-4 (14B), and Qwen 2.5 (7B) are the respective default models from Ollama.[2] For comparison, we also included GPT-4o-mini as a closed-source baseline. Appendix B shows the final prompt we used for submission after preliminary evaluations.

Argumentation scheme integration: We also tested an approach that incorporated argumentation schemes from Walton et al. (2008) into the prompting process. These schemes were embedded into prompts, encouraging the model to generate questions targeting assumptions, analogies, consequences, and other reasoning patterns. Although this method aligned with the theoretical foundations of critical question generation, it was not included in the final submission due to lower empirical performance during our preliminary tests on the validation dataset. Appendix A provides more details on this approach.

4.2 Stage 2: Reranking Critical Questions

Model selection and fine-tuning: To select the most useful questions from the set of generated candidates, we implemented an evaluation and reranking pipeline using fine-tuned classification models. We tested BERT, DistilBERT, and ModernBERT, with ModernBERT demonstrating the best performance in our preliminary evaluations.

While BERT and DistilBERT showed some promise in preliminary evaluations, they struggled to process longer interventions and complex critical questions—possibly due to limited context size. In contrast, ModernBERT performed better, likely because it could handle longer inputs—some interventions exceeded the context size limit of standard BERT models—allowing it to consider a more complete argumentative context during classification.

We fine-tuned two variants of ModernBERT. The first was trained on the validation dataset provided by the organizers. The second variant was further trained on approximately 67.8k critical questions generated by the LLMs listed in the model selection section, using the diverse prompting strategies described in Section 3. These questions were automatically labeled using the official evaluation script, which assigns labels based on semantic similarity with reference questions from the validation set. Questions labeled as `not_able_to_evaluate` were discarded, along with duplicate questions generated across different LLMs and prompts, to reduce redundancy in the fine-tuning dataset. This extended version generalized better across different prompt styles and generation patterns in our preliminary experiments.

Training was conducted for 5 epochs using the AdamW optimizer with a learning rate of 5e-5. Evaluation and checkpoint saving were performed at the end of each epoch, with the best model selected based on the F1-score.

Data preparation: We merged the `Unhelpful` and `Invalid` categories into a single `Non-Useful` class to simplify the classification task, since both receive zero points in the evaluation. The dataset was then split into training, validation, and internal test sets. We fine-tuned ModernBERT to perform binary classification, predicting whether a given critical question is `Useful` or `Non-Useful`. This internal test set was used solely for development and is distinct from the official shared task test set.

Evaluation and reranking: To ensure that the final output included the most useful and diverse critical questions, we combined the outputs of two LLMs. For each intervention, three candidates were generated by each model, resulting in six critical questions. These were scored by ModernBERT per its confidence in the predicted usefulness (0 meaning 100% confident it is not useful), and the

[2] https://ollama.com

top three were selected. This multi-model generation and reranking strategy leveraged the strengths of different LLMs while ensuring output consistency through a unified reranking mechanism.

We submitted three runs for evaluation, selected based on their performance on the official sample and validation sets. The first submission used output from Gemma 2 with a single prompt; no reranking is needed as only three critical questions are generated. The second submission combined outputs from Gemma 2 and Phi-4, reranked using ModernBERT fine-tuned on both validation data and additional generated critical questions (Reranker-Augmented). The third submission used the same prompting setup but reranked with a model trained only on the validation set (Reranker-Base).

5 Results

Table 1 presents the evaluation scores for all systems on the official sample, validation, and test sets. Among the prompting-only models, Gemma2 achieved the highest scores on the sample (0.53) and validation set (0.72). Phi-4, Mistral, Gemma 3, and LLaMA 3.2 showed moderate performance while Qwen 2.5 performed worst.

In contrast, our reranking pipeline significantly improved performance. Reranker-Augmented achieved a test score of 0.57, marking the best result among all our submissions. Reranker-Base, which shared the same LLM generation setup, yielded a slightly lower score of 0.48. These results validate the effectiveness of combining prompt diversity with model-based reranking.

For high-scoring submissions in the automatic evaluation mentioned above, the organizers manually reviewed critical questions that the automated evaluator marked as `not_able_to_evaluate`. For Reranker-Augmented, this included 12 such cases; after manual review, some were relabeled as `Useful`, increasing the total to 58 `Useful` questions, resulting to the final score of 0.57.

It is worth noting that we do not have access to the relabeling outcomes for the other two submissions, which included 12 (Gemma2) and 19 (Reranker-Base) `not_able_to_evaluate` questions. If some of these were similarly reclassified as `Useful`, their final scores would be higher.

Method / Model	**Evaluation Score**		
	Sample	**Validation**	**Test**
Prompting			
Gemma 2*	**0.53**	**0.72**	0.49
Gemma 3	0.27	0.60	
LLaMA 3.2	0.40	0.58	
Mistral	0.33	0.61	
Phi-4	0.33	0.68	
Qwen 2.5	0.27	0.59	
Prompting + Reranking			
Reranker-Augmented*	**0.67**	**0.84**	**0.57**
Reranker-Base*	0.56	0.82	0.48
Argumentation Scheme Integration			
Gemma 2	0.60	0.72	

Table 1: Evaluation scores on the Sample, Validation, and Test sets for Prompting-only and Prompting + Reranking strategies. Stars (*) mark the three systems submitted to the shared task. The score 0.57 on the test set was partially based on manual evaluation.

6 Conclusion

We presented the submission of team *Webis* to the ArgMining 2025 Critical Question Generation shared task. Our system employed a two-stage pipeline combining diverse prompting strategies with a reranking mechanism powered by ModernBERT. Among our submissions, the best-performing configuration achieved a test score of 0.57 and ranked 5th overall.

As the goal of the task was to automate the generation of critical questions, we did not manually edit or post-process any of the outputs. All results were derived directly from the LLMs and the reranking model without human intervention, ensuring complete pipeline automation.

Our results highlight that even relatively lightweight open-weight models like Gemma 2, when paired with a reranking classifier fine-tuned on extended data, can yield competitive performance in challenging generative tasks such as critical question generation. The effectiveness of our approach stems from leveraging the diversity of LLM generations and then selecting questions through a classifier trained on inferential quality.

However, we observed that short or single-sentence interventions often led to overly generic or unhelpful critical questions, as the models had limited argumentative context to build upon. Additionally, the reliance on similarity-based evaluation can undervalue useful questions that diverge lexically from reference examples.

One idea for future work is to explore agent-based iterative generation strategies, where a criti-

cal question generation model and a feedback module interact to improve question quality over multiple rounds. Instruction-tuned models or reinforcement learning setups could also be used to explicitly optimize for the usefulness and specificity of generated questions.

Ethics Statement

We participated in the Critical Question Generation Shared Task using the dataset provided by the organizers, without modifying its content. All experiments were conducted solely for research purposes and in accordance with the ACL Ethics Policy. Our system generates critical questions automatically, but it is not yet suitable for deployment in high-stakes or production environments. The focus of this work is to advance research on critical reasoning and question generation in argumentation settings.

References

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.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4171–4186. Association for Computational Linguistics.

Charles-Olivier Favreau, Amal Zouaq, and Sameer Bhatnagar. 2022. Learning to rank with bert for argument quality evaluation. In *Proceedings of the 35th International FLAIRS Conference (FLAIRS-35)*.

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 2025. In *Proceedings of the 12th Workshop on Argument Mining*.

S. M. Towhidul Islam, S. M. Mehedi Zaman, Vinija Jain, Anku Rani, Vipula Rawte, Aman Chadha, and Amitava Das. 2024. A comprehensive survey of hallucination mitigation techniques in large language models. *arXiv preprint arXiv:2401.01313*.

Siddhartha Jain, Xiaofei Ma, Anoop Deoras, and Bing Xiang. 2024. Lightweight reranking for language model generations. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6960–6984, Bangkok, Thailand. Association for Computational Linguistics.

John Lawrence and Chris Reed. 2019. Argument mining: A survey. *Computational Linguistics*, 45(4):765–818.

Bangzheng Li, Ben Zhou, Fei Wang, Xingyu Fu, Dan Roth, and Muhao Chen. 2023. Deceptive semantic shortcuts on reasoning chains: How far can models go without hallucination? *arXiv preprint arXiv:2311.09702*.

Fabrizio Macagno, Douglas Walton, and Chris Reed. 2017. Argumentation schemes. *Argumentation*, 31(4):529–563.

Nick McKenna, Tianyi Li, Liang Cheng, Mohammad Javad Hosseini, Mark Johnson, and Mark Steedman. 2023. Sources of hallucination by large language models on inference tasks. *arXiv preprint arXiv:2305.14552*.

Chris Reed and Douglas Walton. 2001. Applications of argumentation schemes. *Argumentation*, 15(3):239–255.

Jacky Visser, John Lawrence, Chris Reed, Jean H. M. Wagemans, and Douglas Walton. 2021. Annotating argument schemes. *Argumentation*, 35:101–139.

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

A Argumentation Scheme Integration

We conducted exploratory experiments integrating *argumentation schemes* from Walton et al. (Walton et al., 2008) into the CQ generation process. We began by identifying and extracting the argumentation schemes present in the sample and validation datasets. For each scheme, we retrieved its associated critical questions from Walton's Argumentation Schemes.

At generation time, we designed a structured prompt that included the intervention text, its corresponding argumentation scheme(s), a brief definition of each scheme, and representative critical questions drawn from Walton's work. This prompt guided the LLM to reason within a specific argumentative structure, aiming to produce more targeted critical questions. For this experiment, we used the Gemma 2 model to generate outputs. The full prompt template is shown in Figure 2.

Despite its theoretical alignment, this scheme-aware prompting approach was not included in the final submission. During preliminary evaluation

(scores in Table 1), we observed that it often constrained the model's generative flexibility and led to questions that were overly rigid or templated. In contrast, more general prompting strategies produced more diverse and context-sensitive outputs.

You are an expert in argument analysis and critical reasoning.

Your task is to generate exactly **3 high-quality critical questions** that challenge the argument below.

Argument:
`{intervention_text}`

How to Generate Strong Critical Questions:
Each question must challenge the argument's assumptions, reasoning, evidence, consequences, or alternative solutions.

Relevant Argumentation Schemes & Examples:
Below are the argumentation schemes relevant to this intervention, along with examples of critical questions.

Scheme Name
• Definition: <*scheme definition*>

Good Example:
Argument: *"..."*
Good CQ: *"..."*

Bad Example:
Argument: *"..."*
Bad CQ: *"..."* (Not helpful)

Walton's Critical Questions:
– Walton CQ 1
– Walton CQ 2
...

Final Self-Assessment:
• "Does this question challenge the argument's assumptions, reasoning, evidence, consequences, or alternatives?"
– If yes, keep the question.
– If no, refine it to make it more impactful.

Your Task:

- Generate exactly 3 critical questions.
- Ensure each question closely follows Walton's Critical Questions.
- Do not introduce new topics or concepts not present in the argument.
- Write each question in one line without additional explanation.

Figure 2: Example of a scheme-aware prompt used in our exploratory experiment integrating argumentation schemes. The prompt includes scheme definitions, examples, and Walton-style critical questions to guide LLM generation.

B Final Prompt Used

You are an expert in argument analysis and critical reasoning.

Your task is to generate **exactly 3 critical questions** that should be asked before accepting the argument below.

Argument:
`{text}`

Definition of Critical Questions (CQs):
Critical Questions are inquiries designed to evaluate the strength and validity of an argument by uncovering and examining the assumptions underlying its premises. They serve as tools to assess whether an argument is sound or fallacious by challenging its reasoning, evidence, and potential implications.

How to Construct High-Quality Critical Questions:

- **Challenge the reasoning** – Does the argument's conclusion logically follow from its premises?
- **Challenge the assumptions** – Is the argument relying on hidden assumptions that might be false?
- **Challenge the evidence** – What proof supports the argument's claims?
- **Challenge the consequences** – Could there be unintended side effects of accepting the argument?
- **Challenge alternative explanations** – Are there better explanations or solutions?

Examples of Strong Critical Questions:
Example 1: Argument from Cause to Effect
Argument: *"If people migrate, unemployment rises."*
Good CQ: *"Are there other economic factors that contribute to unemployment apart from migration?"*
Bad CQ: *"What is the history of migration?"* (Not directly relevant)

Example 2: Practical Reasoning
Argument: *"Raising the minimum wage makes the economy fairer, so we should raise it."*
Good CQ: *"Are there alternative policies that could also achieve economic fairness without raising the minimum wage?"*
Bad CQ: *"What is the history of minimum wage policies?"* (Too broad)

Final Self-Assessment:
After generating the 3 critical questions, apply this check to each one:
"Can the answer to this question diminish the acceptability of the argument?"
- If **yes**, keep the question.
- If **no**, refine the question to make it more impactful.

Your Task:

- Generate exactly **3 high-quality critical questions**.
- **Ensure each question directly relates to the given argument** (avoid generic questions).
- Do not introduce new topics or concepts not present in the argument.
- After generating each question, apply the self-assessment check.
- Write each question in one line without any explanation.

Now, generate the 3 critical questions:

Figure 3: Final prompt used for critical question generation.