CausalMinds at SemEval-2026 Task 12: Simple Fine-Tuning with Option Shuffling Outperforms Complex Pipelines for Abductive Event Reasoning

Vidur Gupta, Xiaofei Zhao, Jason Shaye
Stanford University
{vidur,xkzhao,jshaye}@stanford.edu

Abstract

We describe our system for SemEval-2026 Task 12 on Abductive Event Reasoning, which requires identifying plausible direct cause(s) of real-world events. We conduct a systematic evaluation of 23 configurations spanning prompting, retrieval-augmented generation, multi-stage verification, and supervised fine-tuning across models of different scales. Across experiments, we found that fine-tuning GPT-4.1-mini with data augmentation via option shuffling consistently outperformed more complex multi-stage pipelines and larger-model prompting strategies. Our system scores 0.88 on the test dataset, ranking 19th out of 221 submissions, which is only 0.07 away from the highest scoring submission of 0.95. Interestingly, chain-of-thought prompting and multi-stage verification hurt performance compared to simpler baselines. This reinforces that simplicity can outperform complex pipelines. We document these negative results and examine the persistent gap between development (0.991) and test (0.88) scores.

1 Introduction

The Abductive Event Reasoning task in SemEval-2026 Task 12: *Towards Real-World Event Causal Inference for Large Language Models* (Ghosh et al., 2026) aims to analyze LLMs' ability to reason about real-world event causality (Cao et al., 2026). In this task, systems are given an observed event description along with retrieved background documents and four candidate causes. The goal is to identify the correct candidate cause or causes, mirroring how humans can create inferences from noisy evidence and distinguish correlation from causality (Section 2.1). In particular, the dataset has been carefully constructed by SemEval and spans politics, finance, and public emergencies.

Evaluating this task is important because modern LLMs can extract information and generate coherent text effectively, yet they often struggle with real-world causal reasoning. Causal reasoning uses the assumption of cause-and-effect relationships to make judgments, whereas abductive reasoning is the inference of the most likely cause of a given outcome (Cao et al., 2026). Even when equipped with retrieved evidence, models often rely on superficial semantic similarity rather than identifying truly causally relevant information (Wang et al., 2025). For example, an LLM might predict that "ice cream sales cause shark attacks" because "ice cream", "summer", and "shark" are semantically related in text but ignore the common cause of hot weather. As a result, this lack of causal reasoning makes it hard for the system to determine right causes from wrong, leading to errors in abductive reasoning.

We evaluated a variety of approaches, spanning prompting strategies, pipelines, ensembles, and fine-tuning. These methods are described in Section 4.2, with related literature discussed in Section 2.2. Building on insights from the results of our approaches, we introduce *CausalMinds*, a system that utilizes data augmentation and supervised fine-tuning on GPT-4.1-mini. Specifically, the training data is augmented by randomly permuting multiple-choice answer positions and remapping gold labels, which are the correct answers in the dataset. Then, the 1,819 original training examples are combined with the 1,819 shuffled examples (Section 3.1). This shuffling of data removes positional and structural biases, such as memorizing that choice "A" is usually right, and encourages the model to look at the causal relationship between each answer and the question. These newly augmented 3,638 examples are then used to supervise fine-tune GPT-4.1-mini for 3 epochs through OpenAI's API (Section 3.2). In our case, supervised fine-tuning means taking a pre-trained language model and showing many examples of questions paired with their correct answers and then slightly adjusting the model's internal parameters so its pre-

Proceedings of the The 20th International Workshop on Semantic Evaluation (2026), pages 2852–2859

dictions become more accurate. By allowing the model to compare its guess to the known correct answer, we can gradually improve its accuracy over many examples.

Our results revealed both strengths and weaknesses of our simple training-focused approach. Despite no chain-of-thought, retrieval augmentation, or other prompting strategies, our team still managed to score 0.88, which ranked 19th place compared to other teams. The fact that our score plateaued around 0.88 even after trying other strategies suggests that performance may be limited by generalization rather than insufficient tuning or overfitting (Section 6).

2 Background

2.1 Task Setup

For our task, models are evaluated based on their ability to perform causal inference under uncertainty. Unlike standard Natural Language Inference (NLI) or retrieval tasks, Abductive Event Reasoning (AER) requires models to perform abductive reasoning. This requires models to infer the most likely cause of an observed event by utilizing incomplete and often noisy evidence. A recurring challenge with this benchmark is that systems can look very strong on the development set but then drop noticeably on the unseen test set. This "dev-test gap" suggests that the main difficulty is building models that generalize reliably to new events and evidence, instead of fitting decisions to a small development split.

The task is framed as a multiple-choice question format, where each instance provides a target event along with the corresponding context. The event is a short description of a real-world occurrence, while the context consists of documents that combine informative evidence as well as potentially "distracting" documents to test if the model can filter out noise. Each event contains four candidate options, and the task allows for multiple correct options (with one option always being "None of the others are correct causes"). The specific dataset format is in JSONL, where each question has a unique topic_id identifier with associated document sets. Appendix A shows a representative example.

During the scoring process, SemEval evaluates model performance using exact and partial matching with the following evaluation scheme:

$$\text{score} = \begin{cases} 1.0 & \text{if } P = G \text{ (exact match)} \\ 0.5 & \text{if } P \subset G \text{ and } P \neq \emptyset \text{ (partial)} \\ 0.0 & \text{otherwise (wrong or over-predicted)} \end{cases}$$

Split	Instances	Multi-answer
Train	1,819	43%
Development	400	48%
Test	612	N/A

Table 1: Dataset statistics. Test labels were not released.

where G is the gold answer set and P is the predicted set (Cao et al., 2026). Note that the condition for partial credit ensures that single-answer questions are scored as either 1.0 (exact match) or 0.0 (wrong). Table 1 shows the data distribution across training, development, and test splits. The training split is used for model fitting, the development split for configuration changes and model selection, and the test split for final evaluation.

2.2 Related Work

Our approach and experimentation process lies at the intersection of causal reasoning, retrieval-augmented generation (RAG), and model fine-tuning. Causal reasoning has been a significant challenge for advanced LLMs, and benchmarks like CausalBench (Zhou et al., 2024) and CausalEval (Yu et al., 2025) demonstrate how models frequently rely on semantic associations rather than a deep understanding of causality. One common issue in these tasks is the presence of annotation artifacts (Gururangan et al., 2018), in which models utilize superficial patterns in the dataset to achieve high performance scores. To address this issue, we adopted option shuffling during training, following prior work demonstrating that such strategies reduce answer label bias and improve generalization in multiple-choice benchmarks (Hendrycks et al., 2023). Furthermore, duplication has been shown to reduce overfitting by training the model on repeated, label-preserving examples, which allows it to learn that the correct answer is determined by the content of the question rather than the fixed answer label (Chen et al., 2023).

Prompting strategies evaluated in this paper included zero-shot, temporal emphasis, and multi-label predictions (Section 4.2). Zero-shot refers to a model performing a task without any task-specific training examples and is often used as a baseline to help evaluate other strategies (Brown et al., 2020). Temporal emphasis is a prompting strategy designed to make a model pay attention to the order of events in time, which LLMs struggle with in particular (Jain et al., 2023). Multi-label pre-

dictions encourage the LLM to consider multiple causes instead of only one, enabling it to consider all relevant contributing factors in multi-answer instances.

Beyond prompting, we explored multi-stage pipelines with verification, where the model generates initial predictions that are then refined by subsequent steps to reduce errors. We also evaluated ensemble methods, including self-consistency (Wang et al., 2023), which aggregates multiple model outputs to select the most frequent answer, and confidence thresholding, which selects only answers above a certain probability cutoff to improve reliability.

Complementing these strategies, we investigated grounding model outputs in factual evidence via RAG (Lewis et al., 2020). We evaluated both keyword-based RAG, which retrieves documents using exact keyword matches from the input query, and embedding-based RAG, which retrieves semantically similar documents using vector embeddings. While this has shown to benefit model performance, raw document retrieval can introduce "semantic distractors" in the context of event reasoning. Sentences in the documents can share specific words with the event yet describe unrelated effects or incidents compared to actual direct causes. While recent work such as CausalRAG (Wang et al., 2025) attempts to filter these distractors through causal graphs, our experiments revealed a more counter-intuitive finding. Fine-tuning a model alone was enough to show optimal performance even without external context. This suggests that through the training process, the model successfully learned necessary domain knowledge.

Finally, Chain-of-Thought (CoT) prompting is a popular technique in LLM research for eliciting reasoning by encouraging step-by-step logic (Wei et al., 2022). However, error propagation can often disrupt these complex reasoning chains in which an early logical hallucination in the chain invalidates the final conclusion (Peng et al., 2025). This finding is significant because CoT and multi-stage verification (both of which require additional reasoning steps) degraded performance by 4.7% and 74.3%, respectively. Instead, we took a different approach by aligning with the LIMA hypothesis presented by Zhou et al. (2023), which argues that LLMs learn an overwhelming majority of their knowledge during pretraining and that "less is more" for alignment. Thus, we found that strongly fine-tuning a smaller, more efficient model (GPT-4.1-mini) outperformed both prompting larger models such as GPT-4o and fine-tuning the full GPT-4.1 on this task.

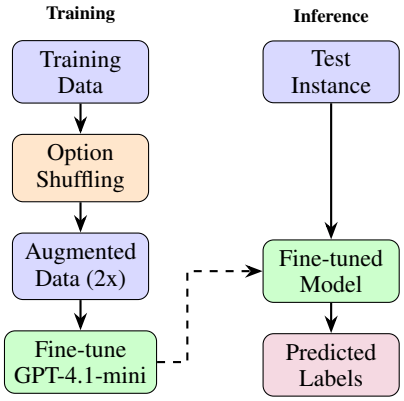

Figure 1: System architecture. Training augments data via option shuffling. At inference, the fine-tuned model directly predicts answer labels without RAG.

Algorithm 1 CausalMinds: Training and Inference

Require: Training set $\mathcal{D} = \{(e_i, O_i, y_i)\}$, test instance (e, O)

Training:

for each $(e, O, y) \in \mathcal{D}$ **do**
  Add (e, O, y) to $\mathcal{D}_{aug}$ ▷ Original instance
  $\pi \leftarrow$ random permutation of $\{A, B, C, D\}$
  Add $(e, \text{Shuffle}(O, \pi), \text{Remap}(y, \pi))$ to $\mathcal{D}_{aug}$
end for
$\mathcal{M} \leftarrow \text{FineTune}(\text{GPT-4.1-mini}, \mathcal{D}_{aug}, \text{epochs}{=}3)$

Inference:

return $\mathcal{M}(e, O)$ ▷ Direct prediction, no RAG

3 System Overview

Figure 1 illustrates our system architecture. We submitted three configurations that all achieved 0.88 on test: fine-tuning with RAG context in training data, fine-tuning with shuffled answer options, and a majority-vote ensemble of three independently fine-tuned GPT-4.1-mini models (base, RAG-augmented, and shuffle-augmented). Since all scored identically on test, we describe the simplest: fine-tuned GPT-4.1-mini with option shuffling augmentation. The provided contextual documents are not used at inference, as we found they introduce distractors (Section 3.3).

Algorithm 1 presents our complete approach, where e is the event description, O is the set of four candidate options, y is the gold label set, and π is a random permutation used to shuffle option positions.

3.1 Data Augmentation

To reduce positional bias, we augment training data by shuffling answer options. For each instance, we generate a random permutation and remap gold labels accordingly, doubling the training data from 1,819 to 3,638 instances. This forces the model to learn causal relationships rather than superficial position patterns (Pezeshkpour and Hruschka, 2024).

3.2 Fine-tuning

We fine-tune GPT-4.1-mini using OpenAI's API with the following configuration: 3 epochs, batch size 1 (API default), learning rate determined automatically by the API, and temperature 0 at inference. See Appendix B for the exact prompt template. We found 3 epochs optimal through ablation: 1 epoch undertrained (0.868) while 5 epochs showed slight overfitting (0.959).

3.3 Retrieval-Augmented Generation

We experimented with keyword-based retrieval-augmented generation (Lewis et al., 2020) that scores document sentences by overlap with event and option keywords, boosting causal indicators. Including RAG context during fine-tuning provided modest gains (+0.4% dev, from 0.965 to 0.969). However, at inference time, the fine-tuned model performed identically with or without RAG context. All three submitted configurations achieved 0.88 on test regardless of RAG usage.

4 Experiments

4.1 Models Evaluated

We tested OpenAI models (GPT-4o, GPT-4.1-mini, GPT-4.1, o1) and Anthropic models (Claude Sonnet 4, Claude Opus 4) across prompting, RAG, and fine-tuning configurations.

4.2 Approaches Explored

We evaluated prompting strategies (zero-shot, chain-of-thought (Wei et al., 2022), temporal emphasis that instructed models to weight temporal ordering of cause and effect, and multi-answer aware prompting that explicitly requested multi-label predictions), retrieval methods (keyword-based RAG (Lewis et al., 2020) and embedding RAG), multi-stage pipelines with verification, ensemble methods including self-consistency (Wang et al., 2023) and confidence thresholding (selecting answers above a probability cutoff), and fine-tuning with different epochs and augmentation strategies.

System	Dev	Test
Claude Opus 4 zero-shot	0.825	–
GPT-4o zero-shot	0.682	–
Claude Sonnet 4 zero-shot	0.590	–
Fine-tuned GPT-4.1-mini	0.965	–
+ RAG (training only)	0.969	0.88
+ option shuffling	0.991	0.88
+ ensemble (3 models)	0.993	0.88
Fine-tuned GPT-4.1 (full)	0.943	–

Table 2: Main results. Our best test score (0.88) ranks 19th.

Approach	Dev	Δ
GPT-4o zero-shot (baseline)	0.682	–
+ chain-of-thought	0.650	−4.7%
+ raw documents	0.620	−9.1%
+ temporal emphasis	0.475	−30.4%
+ multi-stage verify	0.175	−74.3%
OpenAI o1 zero-shot	0.150	−78.0%

Table 3: Approaches that degraded performance.

5 Results

5.1 Main Results

Table 2 shows our main results (full results in Appendix D). The best zero-shot result is Claude Opus 4 at 0.825. Fine-tuning GPT-4.1-mini reaches 0.965, and adding augmentation pushes this to 0.991. On test, we score 0.88 (19th place), 0.07 behind the winning system.

5.2 Negative Results

Table 3 shows what did not work. Chain-of-thought hurt by 4.7%, possibly because explicit reasoning introduces errors that compound. Multi-stage verification hurt by 74.3%, as the verifier often changed correct answers to wrong ones.

5.3 Development-Test Gap Analysis

Despite improving development from 0.965 to 0.993, test performance plateaued at 0.88 across three configurations.

The development set (400 instances) may simply be too small to reliably predict test performance. Yet the fact that three different approaches (RAG, augmentation, ensemble) all hit exactly 0.88 on test, despite dev scores of 0.969, 0.991, and 0.993, points to a model-class ceiling rather than

configuration-specific overfitting. A model-class ceiling occurs when the LLM has inherent limitations, regardless of changes in the configuration, whereas configuration-specific overfitting refers to when poor test performance is due to suboptimal parameters, training choices, or overfitting to the dev set. Additionally, the test set may contain harder instances or different domain proportions.

Without access to test labels, we cannot perform direct error analysis on test predictions. However, the consistency of 0.88 across diverse approaches suggests the gap reflects genuine generalization challenges rather than overfitting to development quirks.

While the large dev–test gap could suggest overfitting, our results provide limited evidence that this gap is driven by memorization of the development set. In particular, configurations that substantially increased development performance (e.g., option shuffling and ensembling, reaching up to 0.993) did not yield any improvement on the test set, where all variants plateaued at 0.88.

6 Analysis

6.1 Error Analysis

We manually analyzed the 14 development instances where our base fine-tuned model erred (see Appendix C). The most frequent error pattern was failing to identify the complete set of contributing causes in multi-answer instances, often predicting only a proper subset of the correct answers. Other recurring errors included temporal ordering confusion and incorrectly selecting "None of the above."

For example, given the event "The district officially unveiled Black Lives Matter Plaza on June 5, 2020," with options including two distinct protest-related causes (A: protests following Floyd's death; B: protests spreading nationwide), city firing officers (C), and video of Floyd's last moments circulating (D), the gold answers were A, B, and D, but our model predicted B and C, missing two contributing causes while also selecting an incorrect one.

6.2 Why Does GPT-4.1 Underperform GPT-4.1-mini?

The larger GPT-4.1 (0.943) underperformed GPT-4.1-mini (0.965). Larger models may be harder to fine-tune on small datasets, having more parameters to adapt with limited signal. The mini model's smaller capacity may act as implicit regularization.

6.3 Why Do Sophisticated Methods Fail?

Task-specific fine-tuning beats sophisticated prompting here. Chain-of-thought likely introduces error propagation when reasoning about causality, and verification models lack task knowledge, compounding rather than catching errors. Although the lack of using contextual documents during inference might be a potential limitation of CausalMinds, our evaluation results comparing the model's performance with vs. without contextual documents (0.62 on Dev) suggests that it was not needed. Raw documents contain distractors that hurt more than they help. OpenAI o1, despite being a strong reasoning model, scored only 0.150, likely because its extended reasoning chains amplify the same error propagation we observe with chain-of-thought, and its output format may not align with our label-only evaluation. Meanwhile, the 1,819 training examples encode patterns that prompting alone cannot capture.

7 Conclusion

CausalMinds scores 0.88 on test (19th place) using fine-tuned GPT-4.1-mini with shuffled answer options. Fine-tuning GPT-4.1-mini (0.965) outperforms the best zero-shot model, Claude Opus 4 (0.825), by 14 points on development, and augmentation via option shuffling pushes this to 0.991. None of the more complex approaches we tried helped. The persistent dev-test gap suggests that future work should prioritize generalization over development set optimization. Notably, our primary contribution is demonstrating that a comparatively simple strategy such as fine-tuning a smaller model (GPT-4.1-mini) combined with lightweight augmentation through option shuffling substantially outperforms more complex, multi-stage pipelines built on larger models. This counterintuitive result suggests that careful task-specific adaptation and robustness-oriented data transformations may yield greater gains than scaling model size or architectural complexity alone.

We have published a GitHub repository showcasing our experiment code and results. [1]

Limitations

One limitation of our approach is its reliance on proprietary models (GPT-4.1-mini via OpenAI's API), which may affect long-term reproducibility if model access, versioning, or pricing policies

[1] https://github.com/vidurgupta01/CausalMinds-SemEval2026-Task12

change. We encourage future work to explore open-weight alternatives, such as Llama-3 or Mistral, to improve replication and strengthen generalizability. Furthermore, we did not perform statistical significance testing due to the limit of submission attempts. The development-test gap suggests findings may not fully generalize to smaller test sets. We did not analyze performance by domain (politics, finance, emergencies), as domain labels were not provided for individual instances. Overall, we believe that we did the best with our given resources for this task.

Acknowledgments

We thank the SemEval-2026 Task 12 organizers for creating this benchmark and our reviewers for their helpful suggestions.

References

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, and 1 others. 2020. Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33:1877–1901.

Pengfei Cao, Yubo Chen, Mingxuan Yang, Chenlong Zhang, Mingxuan Liu, Kang Liu, and Jun Zhao. 2026. SemEval-2026 task 12: Abductive event reasoning: Towards real-world event causal inference for large language models. In *Proceedings of the 20th International Workshop on Semantic Evaluation (SemEval-2026)*. Association for Computational Linguistics.

Jiaao Chen, Derek Tam, Colin Raffel, Mohit Bansal, and Diyi Yang. 2023. An empirical survey of data augmentation for limited data learning in nlp. *Transactions of the Association for Computational Linguistics*, 11:191–211.

Debanjan Ghosh, Kai North, Ekaterina Kochmar, Mamoru Komachi, and Marcos Zampieri, editors. 2026. *Proceedings of the 20th International Workshop on Semantic Evaluation (SemEval-2026)*. Association for Computational Linguistics, San Diego, United States.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman, and Noah A. Smith. 2018. Annotation artifacts in natural language inference data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 107–112, New Orleans, Louisiana. Association for Computational Linguistics.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2023. A study on massive multitask language understanding. *Journal of Machine Learning Research*, 24(79):1–32.

Raghav Jain, Daivik Sojitra, Arkadeep Acharya, Sriparna Saha, Adam Jatowt, and Sandipan Dandapat. 2023. Do language models have a common sense regarding time? revisiting temporal commonsense reasoning in the era of large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6750–6774, Singapore. Association for Computational Linguistics.

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

Jingyu Peng, Maolin Wang, Xiangyu Zhao, Kai Zhang, Wanyu Wang, Pengyue Jia, Qidong Liu, Ruocheng Guo, and Qi Liu. 2025. Stepwise reasoning disruption attack of LLMs. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5040–5058, Vienna, Austria. Association for Computational Linguistics.

Pouya Pezeshkpour and Estevam Hruschka. 2024. Large language models sensitivity to the order of options in multiple-choice questions. In *Findings of the Association for Computational Linguistics: NAACL 2024*. Association for Computational Linguistics.

Nengbo Wang, Xiaotian Han, Jagdip Singh, Jing Ma, and Vipin Chaudhary. 2025. CausalRAG: Integrating causal graphs into retrieval-augmented generation. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 22680–22693. Association for Computational Linguistics.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-consistency improves chain of thought reasoning in language models. In *International Conference on Learning Representations*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837.

Longxuan Yu, Delin Chen, Siheng Xiong, Qingyang Wu, Qingzhen Liu, Dawei Li, Zhikai Chen, Xiaoze Liu, and Liangming Pan. 2025. Causaleval: Towards better causal reasoning in language models.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srini Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, Susan Zhang, Gargi Ghosh, Mike Lewis,

Luke Zettlemoyer, and Omer Levy. 2023. Lima: Less is more for alignment.

Yu Zhou, Xingyu Wu, Beicheng Huang, Jibin Wu, Liang Feng, and Kay Chen Tan. 2024. Causalbench: A comprehensive benchmark for causal learning capability of llms.

A Task Example

Table 4 shows a representative instance from the training set.

Event: Musk fired top Twitter executives.
A. Twitter sued Elon Musk to force the acquisition. B. The Delaware Chancery Court ordered a Friday deadline for the Twitter acquisition deal to close in early October. C. Elon Musk completed his $44 billion takeover of Twitter. D. Elon Musk changed his profile to 'Chief Twit'.
Gold answer: C

Table 4: Example task instance. Only option C is a direct cause: completing the takeover gave Musk the authority to fire executives. Options A and B are temporally prior but not direct causes. Option D is a correlate, not a cause.

B Implementation Details

Fine-tuning hyperparameters. OpenAI API with: 3 epochs, batch size 1 (default), automatic learning rate, no warmup specification. Training completed in approximately 15 minutes.

Inference. Temperature 0, max tokens 32.

RAG configuration. Keyword extraction excludes stop words and requires minimum 4 characters. Causal keywords ("caused," "because," "due to," "led to," "result of," "triggered," "sparked," etc.) receive +5 score boost. Top 3 sentences selected per question.

Paraphrase augmentation. We experimented with paraphrasing answer options using GPT-4o to generate semantically equivalent alternatives. This underperformed option shuffling (0.981 vs 0.991), likely because paraphrasing introduced subtle meaning shifts.

B.0.1 System message

```
You are an expert at causal reasoning. Identify direct causes of events accurately.
```

B.1 Prompt template used for fine-tuning and inference:

```
Identify the direct cause(s) of this event.
Select all options that directly caused the event to happen.

Event: {event}

A. {option_A}
B. {option_B}
C. {option_C}
D. {option_D}

Rules:
- A direct cause must happen BEFORE the event
- There must be a clear causal mechanism
- Multiple options can be correct
- "None of the others" is valid when no option is a true cause

Answer (letters only, comma-separated if multiple):
```

C Error Analysis Details

Table 5 summarizes error patterns from our base fine-tuned model.

Error Pattern	Count
Missing contributing causes in multi-answer	3
Temporal ordering confusion	2
Protest/riot causation errors	2
Incorrect "None of the above" selection	2
Other single-cause errors	5
Total	14

Table 5: Error pattern distribution.

D Complete Experimental Results

Table 6 provides results across all 23 configurations.

Category	Configuration	Dev	Notes
Zero-shot Prompting			
	GPT-4o baseline	.682	Best baseline
	+ chain-of-thought	.650	Reasoning hurts
	+ temporal emphasis	.475	Over-constrained
	+ multi-answer aware	.390	Multi-label hurts
	Claude Sonnet 4	.590	Below GPT-4o
	Claude Opus 4	.825	Best non-finetuning
	OpenAI o1	.150	Reasoning fails
RAG			
	GPT-4o + raw docs	.620	Docs distract
	GPT-4o + keyword RAG	.760	Filtering helps
	GPT-4o + embed. RAG	.750	Similar
Multi-stage Pipelines			
	RAG + reason + verify	.175	Verify hurts
	GPT-4o + Claude verify	.283	Cross-model hurts
Fine-tuning: GPT-4.1-mini			
	1 epoch	.868	Undertrained
	3 epochs	.965	Optimal
	5 epochs	.959	Slight overfit
	3 ep. + RAG context	.969	RAG helps
	3 ep. + shuffled opts	.991	Best single
	3 ep. + paraphrase	.981	Paraphrase hurts
Fine-tuning: GPT-4.1			
	3 epochs	.943	Larger worse
	3 ep. + RAG context	.960	Still worse
Ensemble & Post-proc.			
	3 models voting	.993	Best dev score
	Self-consistency	.980	Below best
	Confidence thresh.	.770	Breaks FT

Table 6: Complete results (23 configurations).