Can LLMs Be Constrained to the Past? Improving Knowledge Cutoff through Recall-Based Prompting

Michiro Asai, Ailiang Lin, Yu Kishimoto, Takao Obi, Satoshi Kosugi, Kotaro Funakoshi, Manabu Okumura
Institute of Science Tokyo
{asai,linailiang,ky,obi,kosugi,funakoshi,oku}@lr.first.iir.isct.ac.jp

Abstract

Prompted knowledge cutoff instructs a large language model (LLM) to act as if information beyond a specified cutoff date were unavailable. However, prior work mainly relies on direct-answer generation, which struggles when post-cutoff knowledge is not explicitly queried but is only causally related to the question. To address this limitation, we propose two recall-based prompting strategies: Self-Recall (SR), which asks the model to restate its cutoff constraint, and Question-Recall (QR), which requires the model to recall question-relevant information valid under the cutoff. Across three existing benchmarks, our methods outperform both direct-answer prompting and conventional step-by-step reasoning baselines, with particularly strong improvements on counterfactual questions. To investigate robustness across different cutoff settings, we further construct the Multi-cutoff Historical Event Benchmark (MHEB), which evaluates the same question under multiple cutoff years. Results show that knowledge cutoff performance varies with cutoff distance, while combining SR and QR consistently yields the best performance. Code and dataset are available at https://github.com/isct-lrlab/MHEB.

1 Introduction

Large language models (LLMs) are increasingly applied to temporal prediction tasks such as stock price forecasting, climate modeling, and traffic prediction (Yu et al., 2023; Jin et al., 2024). However, LLMs are pretrained on large-scale corpora collected over long time spans, raising a fundamental concern of data contamination: evaluation on examples predating the model's actual knowledge cutoff may probe memorized knowledge rather than genuine temporal reasoning (Roberts et al., 2023; Dong et al., 2024). Therefore, constraining an LLM to answer based only on knowledge available before a specified cutoff date is critical to ensuring trustworthy temporal benchmarking and preventing knowledge leakage in real-world applications.

To ensure temporal consistency, prior work[1] typically relies on additional training over curated temporal datasets (Ren et al., 2023; Zhao et al., 2024). While these methods effectively improve temporal grounding, they require internal model access, and thus cannot be applied to black-box LLMs served through APIs. In contrast, Prompted Knowledge Cutoff (Gao et al., 2025) instructs an LLM to unlearn all information beyond a specific cutoff date through prompting, e.g., *"Assume you are an AI with a knowledge cutoff of December 31, 2020."* While this approach is effective when the forgotten information is explicitly queried, its performance degrades significantly in cases where that information is only causally related to the question. Another limitation is that existing prompted-cutoff benchmarks assign a single fixed cutoff date to each question. This setup reveals whether a prompt succeeds at one particular temporal boundary, but not how performance varies as the cutoff is shifted further back from the event year.

To this end, we propose two novel recall-based prompting strategies that generate intermediate outputs to improve knowledge cutoff performance: a **Self-Recall (SR)** prompt that encourages the model to output one short statement of its cutoff stance before answering, recasting the temporal constraint from an external instruction into a self-declared state that is more likely to persist through generation; and a **Question-Recall (QR)** prompt that instructs the model to produce a short overview of question-relevant facts together with their associated years before answering, anchoring retrieval to the specified cutoff and suppressing activation of post-cutoff information. For evaluation, we further construct the **Multi-cutoff Historical Event Benchmark (MHEB)**, in which each instance is

[1] See Appendix A for related work.

Proceedings of the 2026 Conference on Empirical Methods in Natural Language Processing, pages 20622–20637
October 24-29, 2026 ©2026 Association for Computational Linguistics

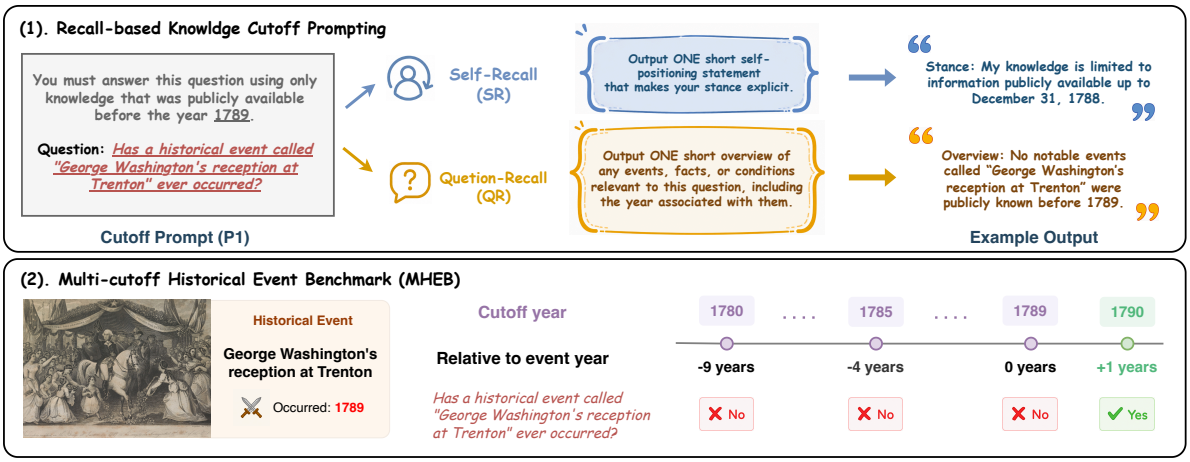


Figure 1: (1) Proposed recall-based prompting strategies: Self-Recall (SR) and Question-Recall (QR), illustrated with *cutoff year* 1789. (2) The proposed Multi-cutoff Historical Event Benchmark (MHEB).

grounded in a historical event from the 13th to 21st century. By pairing the same question with cutoff dates ranging from zero to nine years before the event, we can directly measure how knowledge cutoff performance varies as the cutoff is shifted relative to the event year.

Comprehensive evaluations on the three knowledge cutoff benchmarks of Gao et al. (2025) across three popular LLMs demonstrate that SR, QR, and their combination SR→QR (i.e., applying SR before QR) consistently improve performance over conventional prompting baselines. In addition, we find that existing step-by-step prompting methods such as zero-shot Chain-of-Thought and Plan-and-Solve may significantly degrade knowledge cutoff performance. Experimental results on MHEB further confirm that SR→QR achieves the highest success rate at all ten cutoff offsets from the event year, demonstrating the robustness of our approach.

2 Methodology

2.1 Self-Recall and Question-Recall

While prompting methods that generate intermediate outputs, such as step-by-step reasoning, are known to improve LLM performance on various tasks (Kojima et al., 2022; Wang et al., 2023), prior work (Gao et al., 2025) on prompted knowledge cutoff has only examined direct-answer prompting. We close this gap by introducing two novel recall-based prompting strategies, Self-Recall and Question-Recall, as shown in Figure 1.

Self-Recall (SR). Previous research suggests that prompting a model to first restate its assigned role can improve LLMs' reasoning capabilities (Kong et al., 2024). Inspired by this finding, we propose SELF-RECALL, which aims to strengthen role adherence under prompted knowledge cutoff. SR requires the model to explicitly recall its knowledge cutoff condition before generating the final answer. In this way, we convert the constraint from an external user instruction into a self-declared internal state of the LLM, making the intended temporal boundary more likely to be reflected through subsequent generation.

Question-Recall (QR). QUESTION-RECALL incorporates temporal context by instructing the model to first recall question-relevant facts available at the specified cutoff date. Unlike simple fact enumeration, QR pairs each recalled fact with its temporal information, thereby explicitly organizing the retrieved knowledge with respect to the cutoff boundary. This preliminary recall step is intended to anchor knowledge retrieval at the cutoff and suppress the model's reliance on post-cutoff information.

Prompt Design. Both prompting methods are designed to *control the model's selection of existing knowledge rather than to enhance reasoning*. As illustrated in Figure 1, we restrict the intermediate recall output in both methods to a single short sentence, without imposing additional tasks on the model, such as excessive structuring, detailed explanations, or explicit reasoning. In addition to applying SR and QR individually, we evaluate two ordered combinations, SR→QR and QR→SR, to examine whether jointly combining role reinforcement and temporally grounded recall improves performance.

2.2 Multi-cutoff Historical Event Benchmark

Existing prompted knowledge cutoff benchmarks (Gao et al., 2025) fix each question's cutoff year to the event year itself, leaving the effect of varying the cutoff unexamined. To address this, we construct the **Multi-cutoff Historical Event Benchmark (MHEB)**, which allows us to evaluate the generalization of prompted knowledge cutoff methods across different cutoff year settings.

Construction. We collect historical events from Wikidata, focusing on entities labeled as `war` (Q198), `battle` (Q178561), or `occurrence` (Q1190554), each annotated with an occurrence year. For each candidate event, we ask all evaluated LLMs to identify its occurrence year without any cutoff instruction and retain only events answered correctly by every model. This joint filtering ensures that subsequent cutoff failures cannot be attributed to a lack of factual knowledge and that all models are evaluated on the same set of events. The final dataset contains 900 historical events, with 100 sampled from each century starting from the 13th century (1201–1300).

Format. Each instance is formulated as a binary question, *"Has a historical event called {event} ever occurred?"*. For each event, the cutoff year can be set to varying offsets before the event year. When no cutoff is applied, the correct answer is "Yes"; when the cutoff year is at or before the event year, the correct answer is "No". This design allows us to evaluate whether prompted cutoff remains effective as the cutoff year is shifted farther back from the event year. Further details about MHEB are provided in Appendix B.4.

3 Experiments

3.1 Experimental Setup

Benchmarks. In addition to **MHEB**, we use three datasets from Gao et al. (2025): **Factual** (binary-choice questions on events that reflect meaningful shifts in world state), **Semantic** (free-form definitions of words with shifted meanings over time), and **Counterfactual** (multi-choice questions requiring causal inference about outcomes affected by events after the cutoff).

We use the unlearn success rate as the primary evaluation metric across all datasets. In particular, Counterfactual is originally measured with a `Soft` metric (success if the model's output differs from the answer it would give without the cutoff); we additionally report a stricter `Hard` metric (success if the model selects the correct answer under the cutoff). To ensure cutoff failures reflect retrieval choice rather than missing relevant knowledge, we restrict Factual and Counterfactual to questions that every model answers correctly without a cutoff prompt. More details about these datasets can be found in Appendix B.

Method	Factual	Semantic	Counterfactual	
			Hard	Soft
Llama-3.3-70B-Instruct				
P1	0.681	0.685	0.390	0.514
w/ ZS-CoT	0.708	**0.718**	0.326	0.416
w/ ZS-PS	0.633	0.698	0.364	0.454
w/ SR	0.767	0.695	0.532	**0.655**
w/ QR	0.706	0.681	0.436	0.529
w/ SR→QR	**0.803**	0.688	**0.586**	0.651
w/ QR→SR	0.773	0.644	0.502	0.588
GPT-4o				
P1	0.647	0.668	0.505	0.567
w/ ZS-CoT	0.710	0.654	0.251	0.286
w/ ZS-PS	0.725	0.668	0.317	0.360
w/ SR	0.735	0.685	0.646	**0.772**
w/ QR	0.685	0.621	0.551	0.594
w/ SR→QR	**0.743**	**0.711**	**0.664**	0.740
w/ QR→SR	0.712	0.674	0.597	0.678
gpt-oss-120b				
P1	0.807	0.666	0.616	0.683
w/ ZS-CoT	0.776	0.634	0.612	0.674
w/ ZS-PS	0.782	0.661	0.564	0.626
w/ SR	**0.825**	0.670	0.684	0.776
w/ QR	0.776	0.644	0.689	0.770
w/ SR→QR	0.788	**0.683**	0.702	0.819
w/ QR→SR	0.791	0.646	**0.712**	**0.824**

Table 1: Knowledge cutoff success rates across three LLMs on the three datasets from Gao et al. (2025). For each model–dataset combination, the best value is highlighted in **bold** and the second-best value is underlined.

Models. We evaluate three LLMs: GPT-4o, gpt-oss-120b, and Llama-3.3-70B-Instruct. The temperature is set to 0 for all models. Further implementation details are provided in Appendix C.

Baselines. We use **P1** from Gao et al. (2025) as the base cutoff instruction, which prompts the model to unlearn knowledge after a specified cutoff date. We further compare against two existing step-by-step prompting baselines applied on top of P1: Zero-Shot Chain-of-Thought (**ZS-CoT**) (Kojima et al., 2022), which appends "Let's think step by step" before the answer, and Zero-Shot Plan-and-Solve (**ZS-PS**) (Wang et al., 2023), which prompts the model to first plan and then solve. The full prompt templates used in this work are provided in Appendix D.

Prompt	$Y^{\text{cutoff}} - Y^{\text{event}}$ (Relative cutoff year from the event year)										
	-9	-8	-7	-6	-5	-4	-3	-2	-1	0	avg
P1	0.749	0.738	0.739	0.742	0.723	0.716	0.710	0.713	0.697	0.673	0.720
w/ ZS-CoT	0.824	0.820	0.816	0.811	0.811	<u>0.824</u>	0.809	0.818	<u>0.831</u>	<u>0.858</u>	0.822
w/ ZS-PS	0.770	0.763	0.756	0.748	0.752	0.748	0.757	0.746	0.739	0.768	0.755
w/ SR	<u>0.833</u>	0.830	0.822	0.821	0.818	0.813	0.809	0.812	0.806	0.794	0.816
w/ QR	0.806	0.787	0.761	0.768	0.769	0.787	0.763	0.778	0.779	0.827	0.782
w/ SR → QR	**0.892**	**0.898**	**0.889**	**0.891**	**0.890**	**0.892**	**0.889**	**0.893**	**0.910**	**0.910**	**0.895**
w/ QR → SR	0.831	<u>0.833</u>	<u>0.827</u>	<u>0.834</u>	<u>0.824</u>	<u>0.824</u>	<u>0.829</u>	<u>0.828</u>	<u>0.831</u>	0.852	<u>0.831</u>

Table 2: Knowledge cutoff success rates on the Multi-cutoff Historical Event Benchmark using Llama3.3-70b-Instruct. In each column, the highest success rate is shown in **bold** and the second-highest value is <u>underlined</u>.

3.2 Results on Knowledge Cutoff Datasets

Table 1 reports knowledge cutoff success rates on the benchmarks of Gao et al. (2025). Overall, SR consistently improves over the P1 baseline across all models and datasets. Despite requiring only a short intermediate output, SR also outperforms existing step-by-step prompting baselines in most settings. The improvements are especially clear on the Counterfactual dataset, where SR-based methods achieve the best performance on questions that require causal reasoning under a cutoff. On Counterfactual, SR→QR significantly outperforms P1 for all three models according to exact McNemar tests on the instance-level predictions ($p < 0.005$). We attribute these gains to the explicit recall of the cutoff condition, which helps the model better maintain the intended temporal constraint during answer generation.

In contrast, ZS-CoT and ZS-PS often perform worse than P1 alone, especially on the Counterfactual dataset. We hypothesize that explicit reasoning may encourage the model to retrieve question-related facts that occurred after the cutoff year, causing it to violate the cutoff constraint and revert to the no-cutoff answer. By comparison, our recall-based prompting strategies are designed to guide the model's knowledge selection without eliciting lengthy reasoning, thereby reducing its reliance on post-cutoff information and making causal inference under the cutoff more reliable.

Finally, combining SR and QR further improves performance across multiple models and datasets, with SR→QR generally yielding the best overall results. This suggests that first declaring the cutoff role helps the subsequent question-specific recall operate under a stronger temporal constraint, while QR further grounds the answer in temporally valid evidence and suppresses references to knowledge after the cutoff. We further evaluate SR→QR on reasoning models in Appendix E.1, with consistent improvements on both GPT-5.2 and o3. We additionally analyze inference efficiency in Appendix E.2, showing a favorable performance–cost trade-off compared with step-by-step prompting. Examples of questions and answers for each dataset, together with a case study of intermediate outputs, are provided in Appendix E.3.

3.3 Results on MHEB

Table 2 reports results on MHEB under different cutoff distances, with $Y^{\text{cutoff}} - Y^{\text{event}}$ ranging from -9 to 0. For example, a value of -5 means that the cutoff is set five years before the event year, so the model should answer using only knowledge available before the event occurred.

Although MHEB contains only events whose occurrence years are correctly identified by the model without a cutoff prompt, knowledge cutoff still fails in many cases. This confirms that such failures are not due to missing event knowledge, but rather to the model's difficulty in suppressing knowledge that should be unavailable under the specified cutoff. Notably, SR→QR achieves the best performance across all cutoff settings, indicating that combining role reinforcement with question-specific recall effectively generalizes knowledge cutoff across different temporal distances.

Moreover, we observe different trends across prompting methods. P1 and SR tend to achieve higher success rates as the cutoff moves farther away from the event year, suggesting that a larger temporal gap makes the cutoff constraint easier to maintain. In contrast, step-by-step prompting methods often perform best when $Y^{\text{cutoff}} - Y^{\text{event}} = 0$. Since these methods often generate question-related temporal information during intermediate reasoning, we hypothesize that such temporal cues are useful when the cutoff is close to the event year, but become less reliable as the cutoff is shifted farther back.

4 Conclusion

In this work, we proposed two recall-based prompts, Self-Recall (SR) and Question-Recall (QR), for prompted knowledge cutoff in LLMs. Experiments on three existing benchmarks across three LLMs show that our methods consistently outperform direct-answer and step-by-step prompting, with the largest gains on counterfactual questions where the relevant knowledge is only causally related to the question. To further investigate robustness under different cutoff settings, we introduce the Multi-cutoff Historical Event Benchmark (MHEB). Notably, applying SR before QR achieves the best performance across the full range of cutoff offsets on MHEB. We hope this work offers a new perspective for improving prompted knowledge cutoff in LLMs.

Limitations

Due to computational constraints, our experiments focus on prompt-based methods and do not explore unlearning techniques that require fine-tuning on additional datasets. In addition, all prompts and benchmarks used in this work are in English. Moreover, MHEB is intentionally designed as a controlled benchmark based on binary historical-event questions, and therefore does not cover more complex temporal reasoning settings such as multi-hop reasoning, temporal QA, or open-ended generation. Extending recall-based prompting to multilingual settings and expanding MHEB to more complex temporal reasoning tasks are left for future work.

Acknowledgments

This study was carried out using the TSUBAME4.0 supercomputer at Institute of Science Tokyo.

References

Sandhini Agarwal, Lama Ahmad, Jason Ai, Sam Altman, Andy Applebaum, Edwin Arbus, Rahul K Arora, Yu Bai, Bowen Baker, Haiming Bao, et al. 2025. gpt-oss-120b & gpt-oss-20b model card. *arXiv preprint arXiv:2508.10925*.

Yejin Bang, Ziwei Ji, Alan Schelten, Anthony Hartshorn, Tara Fowler, Cheng Zhang, Nicola Cancedda, and Pascale Fung. 2025. Hallulens: Llm hallucination benchmark. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 24128–24156.

Nora Belrose, David Schneider-Joseph, Shauli Ravfogel, Ryan Cotterell, Edward Raff, and Stella Biderman. 2023. Leace: Perfect linear concept erasure in closed form. *Advances in Neural Information Processing Systems*, 36:66044–66063.

Tom B. Brown, Benjamin Mann, Nick Ryder, et al. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, pages 1877–1901.

Rishav Chourasia and Neil Shah. 2023. Forget unlearning: Towards true data-deletion in machine learning. In *International conference on machine learning*, pages 6028–6073. PMLR.

Bhuwan Dhingra, Jeremy R. Cole, Julian Martin Eisenschlos, Daniel Gillick, Jacob Eisenstein, and William W. Cohen. 2022. Time-aware language models as temporal knowledge bases. *Transactions of the Association for Computational Linguistics*, 10:257–273.

Yihong Dong, Xue Jiang, Huanyu Liu, Zhi Jin, Bin Gu, Mengfei Yang, and Ge Li. 2024. Generalization or memorization: Data contamination and trustworthy evaluation for large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 12039–12050.

Rohit Gandikota, Joanna Materzynska, Jaden Fiotto-Kaufman, and David Bau. 2023. Erasing concepts from diffusion models. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2426–2436.

Xin Gao, Ruiyi Zhang, Daniel Du, Saurabh Mahindre, Sai Ashish Somayajula, and Pengtao Xie. 2025. Can prompts rewind time for LLMs? evaluating the effectiveness of prompted knowledge cutoffs. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 20777–20788, Suzhou, China. Association for Computational Linguistics.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Rem Hida, Masahiro Kaneko, and Naoaki Okazaki. 2024. Social bias evaluation for large language models requires prompt variations. *arXiv preprint arXiv:2407.03129*.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Joel Jang, Dongkeun Yoon, Sohee Yang, Sungmin Cha, Moontae Lee, Lajanugen Logeswaran, and Minjoon Seo. 2023. Knowledge unlearning for mitigating privacy risks in language models. In *Proceedings*

of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 14389–14408. Association for Computational Linguistics.

Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Y. Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, and Qingsong Wen. 2024. Time-LLM: Time series forecasting by reprogramming large language models. In *The Twelfth International Conference on Learning Representations*.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. In *Advances in Neural Information Processing Systems (NeurIPS 2022)*, pages 22199–22213.

Aobo Kong, Shiwan Zhao, Hao Chen, Qicheng Li, Yong Qin, Ruiqi Sun, Xin Zhou, Enzhi Wang, and Xiaohang Dong. 2024. Better zero-shot reasoning with role-play prompting. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 4099–4113, Mexico City, Mexico. Association for Computational Linguistics.

Yein Park, Chanwoong Yoon, Jungwoo Park, Minbyul Jeong, and Jaewoo Kang. 2025a. Does time have its place? temporal heads: Where language models recall time-specific information. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16616–16643, Vienna, Austria. Association for Computational Linguistics.

Yein Park, Chanwoong Yoon, Jungwoo Park, Donghyeon Lee, Minbyul Jeong, and Jaewoo Kang. 2025b. Chroknowledge: Unveiling chronological knowledge of language models in multiple domains. In *International Conference on Learning Representations (ICLR 2025)*.

Han Ren, Hai Wang, Yajie Zhao, and Yafeng Ren. 2023. Time-aware language modeling for historical text dating. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 13646–13656, Singapore. Association for Computational Linguistics.

Manley Roberts, Himanshu Thakur, Christine Herlihy, Colin White, and Samuel Dooley. 2023. To the cutoff... and beyond? a longitudinal perspective on llm data contamination. In *The Twelfth International Conference on Learning Representations*.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. Mpnet: Masked and permuted pre-training for language understanding. In *Advances in Neural Information Processing Systems (NeurIPS 2020)*, pages 16857–16867.

Lei Wang, Wanyu Xu, Yihuai Lan, Zhiqiang Hu, Yunshi Lan, Roy Ka-Wei Lee, and Ee-Peng Lim. 2023. Plan-and-solve prompting: Improving zero-shot chain-of-thought reasoning by large language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2609–2634. Association for Computational Linguistics.

Weiqi Wang, Zhiyi Tian, Chenhan Zhang, and Shui Yu. 2024a. Machine unlearning: A comprehensive survey. *arXiv preprint arXiv:2405.07406*.

Xinlei Wang, Maike Feng, Jing Qiu, Jinjin Gu, and Junhua Zhao. 2024b. From news to forecast: Integrating event analysis in llm-based time series forecasting with reflection. *Advances in Neural Information Processing Systems*, 37:58118–58153.

Xinli Yu, Zheng Chen, and Yanbin Lu. 2023. Harnessing LLMs for temporal data - a study on explainable financial time series forecasting. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 739–753.

Yimeng Zhang, Xin Chen, Jinghan Jia, Yihua Zhang, Chongyu Fan, Jiancheng Liu, Mingyi Hong, Ke Ding, and Sijia Liu. 2024. Defensive unlearning with adversarial training for robust concept erasure in diffusion models. *Advances in neural information processing systems*, 37:36748–36776.

Bowen Zhao, Zander Brumbaugh, Yizhong Wang, Hannaneh Hajishirzi, and Noah Smith. 2024. Set the clock: Temporal alignment of pretrained language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 15015–15040, Bangkok, Thailand. Association for Computational Linguistics.

A Related Work

A.1 Ensuring Temporal Consistency in LLMs

LLMs are trained on corpora spanning long time periods, which can lead to temporally inconsistent knowledge recall. For example, given the question "In 2019, in what year was the Tokyo Olympics scheduled to be held?", a temporally consistent model should answer "2020", but may instead recall the post-postponement outcome and answer "2021". This mismatch arises when the time assumed by the question differs from the time point of the knowledge used by the model.

Several studies have explored ways to align LLM responses with a target time. Dhingra et al. (2022) pretrain models on timestamped corpora and show that models can represent "time-conditioned knowledge". Zhao et al. (2024) align model knowledge to a desired year through fine-tuning on temporally curated data. Park et al. (2025a) identify attention heads specialized for recalling time-dependent knowledge, suggesting that temporal knowledge may be editable through internal interventions. However, these methods require either additional training or internal model modifications, making them difficult to apply to closed-source LLMs served only through APIs. In contrast, Park et al. (2025b) propose ChroKnowPrompt, a prompting method that aligns the model's knowledge recall with a specific year by using correct facts from neighboring years as few-shot examples (Brown et al., 2020). While this avoids parameter updates, it assumes access to reliable facts around the target year, which may not always be available.

A.2 Machine Unlearning

Machine unlearning aims to make a trained model forget specific knowledge, often for privacy or safety reasons (Wang et al., 2024a). Prior work has studied removing the influence of selected training examples (Chourasia and Shah, 2023; Jang et al., 2023; Wang et al., 2024b) or suppressing specific concepts from model representations (Belrose et al., 2023; Gandikota et al., 2023; Zhang et al., 2024). However, most unlearning methods require access to model parameters or additional training, making them difficult to apply to black-box LLMs.

Prompted Knowledge Cutoff (Gao et al., 2025) offers a lightweight alternative by using prompts to make an LLM behave as if it had forgotten information after a specified cutoff year. This framework requires neither parameter updates nor model-internal access, making it practical for black-box LLMs. Gao et al. (2025) evaluate prompted cutoff behavior on three settings: Factual, which tests direct facts whose answers change over time; Semantic, which tests words whose meanings shift across years; and Counterfactual, which tests questions where post-cutoff events causally affect the answer. Their results show that prompted cutoff is relatively effective when the target knowledge is directly queried, but degrades substantially in Counterfactual settings where the post-cutoff knowledge is only causally related to the question. Our work builds on this finding by introducing recall-based prompting strategies and a multi-cutoff benchmark to better evaluate robustness across different cutoff year settings.

B Benchmark Details

This section provides additional details on the four benchmarks used in our experiments: the three datasets from Gao et al. (2025), which are distributed under the Apache License 2.0, and our Multi-cutoff Historical Event Benchmark (MHEB).[2] The Factual, Semantic, Counterfactual, and MHEB datasets contain 675, 303, 689, and 900 examples, respectively.

B.1 Factual

The model is asked a binary-choice question about direct factual changes in the state of the world, such as a person, institution, or event at a particular year. The evaluation checks whether the model's final answer matches the correct answer under the specified cutoff year. One option corresponds to the correct answer without applying a knowledge cutoff (Answer before unlearn), while the other corresponds to the correct answer after applying the cutoff (Answer after unlearn). The knowledge cutoff success rate is the proportion of cases in which the model selects the cutoff-consistent answer. A example from the Factual dataset is shown below:

Example of the Factual dataset.

"Question": "Is TikTok accessible in India?",
"Answer before unlearn": "No",
"Answer after unlearn": "Yes",
"Cutoff year": "2020"

[2] https://github.com/isct-lrlab/MHEB

B.2 Semantic

The model is asked to define words whose meanings have changed over time, such as words with semantic shifts or neologisms. The evaluation measures whether the model can produce the meaning corresponding to the cutoff year (Meaning after unlearn) under the knowledge cutoff setting. Unlike Factual, the model generates a free-form textual definition of the given word. We evaluate semantic similarity between the model output and the prepared ground-truth definitions using MPNet sentence embeddings (Song et al., 2020). Let y_b and y_a denote the embeddings of the correct word senses before and after the cutoff year, respectively, and let o_b and o_a denote the embeddings of the model outputs before and after applying the knowledge cutoff. A knowledge cutoff is considered successful if the following condition holds:

$$\frac{\cos(o_a,y_a)}{\cos(o_a,y_a)+\cos(o_a,y_b)} > \frac{\cos(o_b,y_a)}{\cos(o_b,y_a)+\cos(o_b,y_b)} \quad (1)$$

where $\cos(\cdot,\cdot)$ denotes cosine similarity. This condition indicates that applying the knowledge cutoff makes the model output relatively closer to the word sense that was valid before the cutoff year. An example from the Semantic dataset is shown below:

Example of the Semantic dataset.

"Word": "TikTok",
"Meaning before unlearn": "A popular video-focused social media platform.",
"Meaning after unlearn": "The mimic of the sound of a clock.",
"Cutoff year": "2016"

B.3 Counterfactual

The model is asked to infer the outcome that would have occurred if it did not know unexpected post-cutoff events, such as postponements, cancellations, or external shocks. Each question is given in a four-choice format. The evaluation checks whether the model's final answer matches the correct answer under the knowledge state at the cutoff year. The options include both the answer without applying a knowledge cutoff (Answer before unlearn) and the answer after applying the cutoff (Answer after unlearn). We report two success rates: *Hard*, which counts a case as successful only when the model selects the correct cutoff-consistent answer, and *Soft*, which counts a case as successful when the model does not select the no-cutoff answer. An example from the Counterfactual dataset is shown below.

Example of the Counterfactual dataset.

"Question": "Which country hosted the 2015 Africa Cup of Nations?",
"Options": {"A": "Equatorial Guinea", "B": "Morocco", "C": "South Africa", "D": "Nigeria"},
"Answer before unlearn": "A",
"Answer after unlearn": "B",
"Cutoff year": "2010"

For the Factual and Counterfactual datasets, we additionally verify whether each model can correctly answer every question without a cutoff prompt. If any model fails on a question under this no-cutoff setting, we exclude that question from evaluation. This ensures that cutoff failures reflect the model's difficulty in suppressing post-cutoff knowledge rather than a lack of relevant factual knowledge.

B.4 MHEB Details

The existing prompted knowledge cutoff benchmark (Gao et al., 2025) evaluates each question under a single cutoff year, making it difficult to study how performance changes across different cutoff settings. To address this limitation, we construct the Multi-cutoff Historical Event Benchmark (MHEB), which evaluates the same question under multiple cutoff years.

We first collect historical events from Wikidata, focusing on event-related entities categorized as `war` (Q198), `battle` (Q178561), or `occurrence` (Q1190554). Each record consists of an event name and its occurrence year. To ensure that the target model already knows the event, we ask it to identify the occurrence year of each candidate event without any cutoff instruction, using the prompt shown below. We retain only events for which the model correctly predicts the year.

Prompt for verifying the event year.

You are a helpful assistant that answers questions about history.
When asked about a year, you MUST answer only with the four-digit year, with no other text.
In what year did {event} occur (or begin)? Please answer with the four-digit year only.

Using the retained events, we construct MHEB

instances following the Factual benchmark format of Gao et al. (2025), as shown below. Each instance asks whether a historical event has ever occurred, with {event} and {event year} replaced by the corresponding Wikidata event name and occurrence year.

MHEB instance format.

"Question": "Has a historical event called {event} ever occurred?"
"Answer before unlearn": "Yes"
"Answer after unlearn": "No"
"Answer switch year": {event year}

The final dataset contains 900 historical events, with 100 events sampled from each century starting from the 13th century (1201–1300). For each event, we evaluate cutoff years from the event year back to nine years before the event year. The correct answer is "Yes" when no cutoff is applied or when the cutoff year is later than the event year, and "No" when the cutoff year is the same as or earlier than the event year.

C Implementation Details

We evaluate three LLMs in our experiments: GPT-4o[3] (Hurst et al., 2024), gpt-oss-120b[4] (Agarwal et al., 2025) with MXFP4 quantization, and Llama-3.3-70B-Instruct[5] (Grattafiori et al., 2024) with BF16 weight precision. GPT-4o is accessed through the OpenAI API[6], gpt-oss-120b through the Together API[7], and Llama-3.3-70B-Instruct is run locally on NVIDIA H100 96GB GPUs. For all models, we set the temperature to 0 and use a fixed random seed when applicable.

D Prompt Templates

This section lists the prompt templates used in our experiments. All methods share the same base cutoff instruction, P1, from Gao et al. (2025). Gao et al. (2025) also designed P2 to encourage the model to impose knowledge constraints during its reasoning process while explicitly prohibiting reasoning explanations in the output. However, P2 was reported to show performance trends similar to P1, with P1 slightly outperforming P2. We confirmed this through preliminary experiments on a subset of our data and therefore conducted all full-data experiments with P1 only.

The templates for P1, ZS-CoT, ZS-PS, SR, QR, SR→QR, and QR→SR are shown in Figures 2–8. Here, {cutoff year} denotes the specified cutoff year, meaning that the model is allowed to use only knowledge available up to December 31 of the previous year. The variable {response format} specifies the required output format for each benchmark: "the most reasonable 'Yes' or 'No"' for the Factual dataset and the proposed MHEB, "a concise, factual definition of the given word or phrase" for the Semantic dataset, and "the letter of the most reasonable answer (A, B, C, or D)" for the Counterfactual dataset.

	P1		SR→QR	
Model	Soft	Hard	Soft	Hard
GPT-4o	0.567	0.505	0.740	0.664
gpt-oss-120b	**0.683**	**0.616**	**0.819**	**0.702**
GPT-5.2	0.494	0.396	0.667	0.570
o3	0.624	0.557	0.664	0.607

Table 3: Knowledge cutoff performance on the Counterfactual benchmark for reasoning and non-reasoning models.

E Additional Results

E.1 Evaluation on Reasoning Models

To examine whether recall-based prompting remains effective for reasoning models, we additionally evaluate SR→QR on GPT-5.2 and o3 using the latest version of the Counterfactual benchmark (Gao et al., 2025). We compare them with GPT-4o and gpt-oss-120b under the same evaluation setting.

As shown in Table 3, reasoning models do not consistently outperform non-reasoning models in our setting. Under P1, GPT-5.2 performs substantially worse than gpt-oss-120b, while o3 under P1 also remains below GPT-4o and gpt-oss-120b with SR→QR. These results suggest that reasoning capability alone does not reliably solve prompted knowledge cutoff.

Importantly, SR→QR remains effective on both reasoning models. It improves GPT-5.2 by 17.4 Hard points and 17.3 Soft points, and improves o3 by 5.0 Hard points and 4.0 Soft points. These results show that SR and QR provide benefits be-

[3] https://platform.openai.com/docs/models/gpt-4o
[4] https://huggingface.co/openai/gpt-oss-120b
[5] https://huggingface.co/meta-llama/Llama-3.3-70B-Instruct
[6] https://openai.com/index/openai-api/
[7] https://www.together.ai/

Method	Hard	Tokens	Latency (s)
P1	0.505	206.7	0.64
w/ ZS-CoT	0.251	408.7	2.52
*w/*ZS-PS	0.317	435.8	2.32
*w/*SR	0.646	262.0	0.81
*w/*QR	0.551	288.2	0.90
*w/*SR→QR	0.664	332.6	1.01

Table 4: Inference efficiency on the Counterfactual dataset using GPT-4o. Hard denotes the strict cutoff success rate. Tokens and wall-clock latency are averaged per question.

yond generic inference-time reasoning by explicitly reinforcing the cutoff constraint and temporally grounding question-relevant recall.

E.2 Efficiency Analysis

To quantify the inference overhead of recall-based prompting, we evaluate token usage and wall-clock latency on the full Counterfactual dataset using GPT-4o. We report average per-question token usage from official API statistics and average wall-clock latency, with temperature set to 0.

As shown in Table 4, compared with P1, SR→QR increases token usage by 61% and latency by 58%, while improving the Hard score by 15.9 points. Compared with the step-by-step baselines, SR→QR uses 19–24% fewer tokens and achieves 57–60% lower latency than ZS-CoT and ZS-PS, while achieving substantially higher accuracy. In particular, SR provides a more efficient alternative, improving the Hard score by 14.1 points with only 27% additional token usage and latency over P1. Overall, our recall-based prompting methods provide a favorable performance–cost trade-off.

E.3 Case Study of Intermediate Outputs

This section presents example outputs from all evaluated prompting methods. The Factual, Semantic, and Counterfactual examples are shown in Tables 5, 6, and 7, respectively.

In particular, Table 7 illustrates how different prompting strategies affect intermediate generation on a Counterfactual example. Both ZS-CoT and ZS-PS explicitly retrieve the post-cutoff fact that the YouTube Partner Program was launched in 2007, leading them to violate the 2005 cutoff and select the incorrect answer. In contrast, SR explicitly reinforces the cutoff stance, while QR recalls question-relevant information consistent with the cutoff. Their combination, SR→QR, incorporates both effects and produces the cutoff-consistent answer. This example illustrates the complementary roles of SR and QR in reducing reliance on post-cutoff knowledge during generation.

F Ethical Considerations

This work studies prompt-based strategies for enhancing knowledge cutoff behavior in LLMs. The datasets used in our experiments are either released by Gao et al. (2025) or constructed from publicly available Wikidata entries, which helps reduce potential ethical concerns to some extent.

Our methods rely on the LLM's own underlying knowledge and therefore may inherit its biases or hallucinations (Hida et al., 2024; Bang et al., 2025). In addition, the proposed recall-based prompting can reduce post-cutoff leakage but cannot guarantee true forgetting. We therefore encourage users to apply our method with caution, especially in settings where reliable information control is required.

P1

You must answer this question using **only knowledge that was publicly available before the year {cutoff year}**. Specifically, your memory ends on **December 31, {cutoff year - 1}**, and you have no access to anything that occurred in {cutoff year} or afterward.

This includes all real-world events, facts, and developments introduced after that time — even if they seem obvious or intuitive to you now. You must assume **ou have never learned them**.

Think silently. Then respond with only {response format} based on what you knew before {cutoff year}.

Do not include any explanation, context, or additional text, just {response format}.

Figure 2: Base cutoff prompt P1 from Gao et al. (2025).

Zero-Shot Chain-of-Thought (ZS-CoT)

You must answer this question using **only knowledge that was publicly available before the year {cutoff year}**. Specifically, your memory ends on **December 31, {cutoff year - 1}**, and you have no access to anything that occurred in {cutoff year} or afterward.

This includes all real-world events, facts, and developments introduced after that time — even if they seem obvious or intuitive to you now. You must assume **ou have never learned them**.

Task:
1. Let's think step by step to solve the question.
2. Answer in the form of {response format}.

Output format:
Reasoning: <output the step-by-step reasoning>
Answer: X
where X is exactly {response format}.

Do not include any other text.

Figure 3: Zero-Shot Chain-of-Thought (ZS-CoT) prompt.

Zero-Shot Plan-and-Solve (ZS-PS)

You must answer this question using **only knowledge that was publicly available before the year {cutoff year}**. Specifically, your memory ends on **December 31, {cutoff year - 1}**, and you have no access to anything that occurred in {cutoff year} or afterward.

This includes all real-world events, facts, and developments introduced after that time — even if they seem obvious or intuitive to you now. You must assume **ou have never learned them**.

Task:
1. Let's first understand the quesiton and devise a plan(2–5 steps) to solve the question.
2. Let's carry out the plan and solve the question step by step.
3. Answer in the form of {response format}.

Output format:
Plan: <output the plan part>
Solve: <output the solve part>
Answer: X
where X is exactly {response format}.

Do not include any other text.

Figure 4: Zero-Shot Plan-and-Solve (ZS-PS) prompt.

Self-Recall (SR)

You must answer this question using **only knowledge that was publicly available before the year {cutoff year}**. Specifically, your memory ends on **December 31, {cutoff year - 1}**, and you have no access to anything that occurred in {cutoff year} or afterward.

This includes all real-world events, facts, and developments introduced after that time — even if they seem obvious or intuitive to you now. You must assume **ou have never learned them**.

Task:
1. Output ONE short self-positioning statement that makes your stance explicit.
2. Answer in the form of {response format}.

Output format:
Stance: <one short sentence explicitly stating the knowledge cutoff and restriction>
Answer: X
where X is exactly {response format}.

Do not include any other text.

Figure 5: Self-Recall (SR) prompt.

Question-Recall (QR)

You must answer this question using **only knowledge that was publicly available before the year {cutoff year}**. Specifically, your memory ends on **December 31, {cutoff year - 1}**, and you have no access to anything that occurred in {cutoff year} or afterward.

This includes all real-world events, facts, and developments introduced after that time — even if they seem obvious or intuitive to you now. You must assume **ou have never learned them**.

Task:
1. Output ONE short overview of any events, facts, or conditions relevant to this question, including the year associated with them (If no notable events occurred before {cutoff year}, simply state that).
2. Answer in the form of {response format}.

Output format:
Overview: <one short sentence highlighting what's relevant to this question>
Answer: X
where X is exactly {response format}.

Do not include any other text.

Figure 6: Question-Recall (QR) prompt.

Self-Recall → Question-Recall (SR → QR)

You must answer this question using **only knowledge that was publicly available before the year {cutoff year}**. Specifically, your memory ends on **December 31, {cutoff year - 1}**, and you have no access to anything that occurred in {cutoff year} or afterward.

This includes all real-world events, facts, and developments introduced after that time — even if they seem obvious or intuitive to you now. You must assume **ou have never learned them**.

Task:
1. Output ONE short self-positioning statement that makes your stance explicit.
2. Output ONE short overview of any events, facts, or conditions relevant to this question, including the year associated with them (If no notable events occurred before {cutoff year}, simply state that).
3. Answer in the form of {response format}.

Output format:
Stance: <one short sentence explicitly stating the knowledge cutoff and restriction>
Overview: <one short sentence highlighting what's relevant to this question>
Answer: X
where X is exactly {response format}.

Do not include any other text.

Figure 7: Self-Recall → Question-Recall (SR → QR) prompt.

Question-Recall → Self-Recall (QR → SR)

You must answer this question using **only knowledge that was publicly available before the year {cutoff year}**. Specifically, your memory ends on **December 31, {cutoff year - 1}**, and you have no access to anything that occurred in {cutoff year} or afterward.

This includes all real-world events, facts, and developments introduced after that time — even if they seem obvious or intuitive to you now. You must assume **ou have never learned them**.

Task:
1. Output ONE short overview of any events, facts, or conditions relevant to this question, including the year associated with them (If no notable events occurred before {cutoff year}, simply state that).
2. Output ONE short self-positioning statement that makes your stance explicit.
3. Answer in the form of {response format}.

Output format:
Overview: <one short sentence highlighting what's relevant to this question>
Stance: <one short sentence explicitly stating the knowledge cutoff and restriction>
Answer: X
where X is exactly {response format}.

Do not include any other text.

Figure 8: Question-Recall → Self-Recall (QR → SR) prompt.

Question, answer and cutoff year	Outputs under different prompts
Question: Does El Salvador adopt Bitcoin as legal tender? **Answer before unlearn:** Yes **Answer after unlearn:** No **Cutoff year:** 2021	**P1:** Yes **P1 w/ ZS-CoT :** On June 9, 2021, the Legislative Assembly of El Salvador voted to pass a bill making Bitcoin legal tender, but since my knowledge cutoff is December 31, 2020, I do not have information about events that occurred after this date. However, I can tell you that as of my knowledge cutoff in 2020, there was no publicly available information that indicated El Salvador had adopted Bitcoin as legal tender. Answer: No **P1 w/ ZS-PS:** Plan: 1. Identify the context of the question, which involves El Salvador and Bitcoin.2. Recall any significant events or announcements related to El Salvador and Bitcoin before December 31, 2020.3. Determine if there were any official statements or actions from El Salvador's government regarding the adoption of Bitcoin as legal tender.4. Evaluate the likelihood of such an event occurring based on available information up to December 31, 2020.5. Conclude based on the information gathered.Solve: Given the steps outlined, we proceed to solve the question:- As of December 31, 2020, there was no publicly available information indicating that El Salvador had adopted Bitcoin as legal tender.- The concept of a country adopting a cryptocurrency as legal tender was highly speculative and not commonly discussed in mainstream economic or political circles at that time.- Without specific information or an official announcement from El Salvador's government, it's reasonable to conclude that as of December 31, 2020, El Salvador had not adopted Bitcoin as legal tender.Answer: No **P1 w/ SR:** Stance: My knowledge is restricted to events and information available up to December 31, 2020. Answer: No **P1 w/ QR:** Overview: In 2020, there was no public information about El Salvador adopting Bitcoin as legal tender, but plans were discussed. Answer: No **P1 w/ SR → QR:** Stance: My knowledge is restricted to events and facts prior to December 31, 2020.Overview: In 2001, El Salvador adopted the US dollar as its official currency, but there were discussions about alternative currencies. Answer: No **P1 w/ QR → SR:** Overview: In 2020, there was no public information about El Salvador adopting Bitcoin as legal tender, but discussions about cryptocurrency adoption were ongoing globally. Stance: My response is based on knowledge available up to December 31, 2020, without access to events or facts introduced after that date. Answer: No

Table 5: Factual benchmark example with outputs from all evaluated prompting methods.

Word, meaning and cutoff year	Outputs under different prompts
Word: flash mob **Meaning before unlearn:** a group of people summoned (as by email or text message) to a designated location at a specified time to perform an indicated action before dispersing **Meaning after unlearn:** This word has no clear definition in the dictionary **Cutoff year:** 2003	**P1:** A flash mob is a group of people who assemble suddenly in a public place, perform an unusual or seemingly pointless act for a brief time, then quickly disperse, often organized via telecommunications, social media, or viral emails. **P1 w/ ZS-CoT :** 1. Consider the time frame: the definition must be based on knowledge available before 2003.2. “Flash mob" is a term that emerged in the early 2000s.3. A flash mob typically involves a group of people who assemble suddenly in a public place, perform an unusual or seemingly pointless act for a brief time, then quickly disperse.4. The concept was popularized through the internet and social media, which were gaining traction in the early 2000s. Answer: A flash mob is a group of people who assemble suddenly in a public place to perform an unusual act for a brief time and then quickly disperse, often organized via the internet. **P1 w/ ZS-PS:** Plan: 1. Identify the key components of the term “flash mob."2. Define the term based on its components and known usage before 2003.Solve:1. A “flash mob" involves a group of people.2. The group assembles suddenly in a public place.3. The purpose is to perform an unusual or seemingly pointless act for a brief time.4. The group disperses quickly after the act. Answer: A flash mob is a group of people who assemble suddenly in a public place to perform an unusual or seemingly pointless act for a brief time and then disperse. **P1 w/ SR:** Stance: My knowledge is limited to information available before 2003. Answer: A flash mob is a group of people who assemble suddenly in a public place, perform an unusual or seemingly pointless act for a brief time, then quickly disperse, often organized via telecommunications, social media, or viral emails. **P1 w/ QR:** Overview: The term “flash mob" gained popularity in the early 2000s, particularly after the first widely recognized flash mob event in 2003. Answer: A flash mob is a group of people who assemble suddenly in a public place, perform an unusual or seemingly pointless act for a brief time, then quickly disperse, often organized via telecommunications, social media, or viral emails. **P1 w/ SR → QR:** Stance: My knowledge is limited to information available before 2003. Overview: The term “flash mob" was coined in 2003, so it is outside my knowledge base. Answer: I cannot provide a definition for “flash mob" as it was not a recognized term before 2003. **P1 w/ QR → SR:** Overview: The term “flash mob" was coined in 2003, after my knowledge cutoff. Stance: I cannot provide a definition for “flash mob" as it was introduced after my knowledge cutoff. Answer: I cannot provide a definition for “flash mob" as it was introduced after my knowledge cutoff.

Table 6: Semantic benchmark example with outputs from all evaluated prompting methods.

Question, options, answer and cutoff year	Outputs under different prompts
Question: When did the YouTube Partner Program launch? **Options:** “A": “2007" “B": “2009" “C": “2006" “D": “Never launched" **Answer before unlearn:** A **Answer after unlearn:** D **Cutoff year:** 2005	**P1:** A **P1 w/ ZS-CoT:** YouTube was founded in 2005 and began allowing content creators to earn revenue through its Partner Program a couple of years later. The program was announced and launched in 2007, enabling creators to share ad revenue. Therefore, among the given options, the correct year is 2007. Answer: A **P1 w/ ZS-PS:** Plan: Identify the launch year of the YouTube Partner Program based on pre-2005 knowledge; Recall that it was introduced after YouTube’s founding, around 2007. Solve: The YouTube Partner Program was announced and began in 2007. Answer: A **P1 w/ SR:** Stance: My knowledge is limited to information publicly available up to December 31, 2004. Answer: D **P1 w/ QR:** Overview: No notable events regarding the YouTube Partner Program launch occurred before 2005. Answer: D **P1 w/ SR→QR:** Stance: My knowledge is limited to information available up to December 31, 2004. Overview: The YouTube Partner Program was not launched before 2005. Answer: D **P1 w/ QR→SR:** Overview: As of 2004, YouTube had not introduced a Partner Program. Stance: My knowledge is limited to information publicly available up to December 31, 2004. Answer: D

Table 7: Counterfactual benchmark example with outputs from all evaluated prompting methods.