Parallel Universes for Temporal Evaluation and Knowledge Updating in LLMs

Jonathan Zheng[♣], **Zirui Shao**[♢*], **Alan Ritter**[♣], **Wei Xu**[♣]
[♣]Georgia Institute of Technology, [♢]Zhejiang University
jzheng324@gatech.edu

Abstract

Large language models (LLMs) rely on static pretraining corpora, resulting in outdated knowledge over time. Existing approaches for evaluating knowledge edits either suffer from rapid contamination or rely on counterfactual edits that conflict with rigid existing knowledge. In this work, we propose a simulation-driven approach for evaluating and training LLMs on knowledge insertions. We introduce PARALLELEVENTS, a benchmark that simulates coherent trajectories of fictional yet realistic future events, enabling controlled evaluation without data contamination or conflicts with existing knowledge. We further propose SYNAPSE, a training framework that leverages model-generated data to integrate new knowledge into model parameters via instruction tuning. This synthetic pipeline enables scalable knowledge integration without costly human-curated data. SYNAPSE improves QA accuracy by 14.23 points over prior methods, indicating that simulation-based synthetic training yields robust, coherent knowledge insertion.

1 Introduction

Large language models (LLMs) are increasingly relied upon as general-purpose information sources. However, their knowledge is largely shaped by static training corpora, which become outdated as the world changes, leading to hallucinations, refusals, or stale responses [30, 61]. Existing approaches to evaluating temporal knowledge in LLMs have largely focused on isolated factual changes, such as newly introduced entities [54, 45] or government leadership transitions [77]. These settings capture only narrow forms of knowledge change and are vulnerable to data contamination as the corresponding facts enter subsequent training corpora, limiting controlled comparisons across models and studies over time. To address these limitations, we introduce a simulation-driven framework for studying knowledge insertion in LLMs, using internally consistent fictional worlds for contamination-resistant evaluation.

We further investigate how synthetic data generated from these parallel universes can be used to train models to acquire new knowledge while retaining existing knowledge. Existing approaches to knowledge updating each have their own shortcomings. In-context learning and retrieval-based methods inject new information at inference time [78, 77], but the injected information may conflict with parametric knowledge or interact poorly with learned abstention behaviors. Model editing directly modifies model parameters to incorporate targeted facts [39, 40], but can introduce unintended side effects and scale poorly to large collections of updates. Fine-tuning and continued pre-training offer a more direct way to integrate new knowledge into model parameters [56, 70], but typically depend on continually refreshed training data and can be costly to sustain. These problems motivate studying not only how to evaluate knowledge insertion under controlled conditions, but also how to generate training data that supports effective knowledge acquisition and retention.

[*]Work done as a visiting student at the Georgia Institute of Technology.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

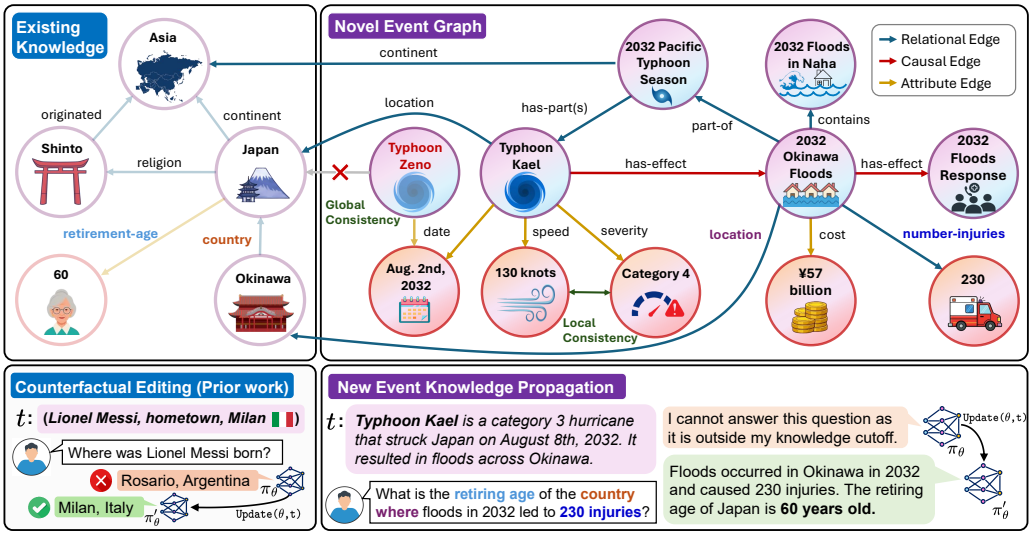

Figure 1: Overview of PARALLELEVENTS, a parallel universe with fictional yet realistic future events up to 2035, for evaluating model insertions under a temporal shift. Factual triples t are categorized into relational, causal, and attribute edges. We enforce local consistency (e.g., wind speed matches disaster severity in a single event) and global consistency for the entire knowledge graph (e.g., no two typhoons occur at the same place and time), enabling rigorous evaluation of knowledge insertions.

More concretely, synthetic environments offer a practical alternative to human-curated corpora, enabling data generation at scale while simulating a changing world. In this work, we leverage these advantages to address two challenges: (1) **benchmark contamination**, where real-world entities and facts enter subsequent training corpora, and (2) **counterfactual brittleness**, where falsified facts conflict with existing commonsense knowledge. However, automated generation can introduce inconsistencies, motivating a hybrid approach with human verification. To this end, we introduce PARALLELEVENTS (Figure 1), a benchmark of plausible yet unseen future events across domains such as sports and natural disasters (§3). These events are extensively verified to ensure local and global consistency. Unlike isolated edits in prior work, these events involve interconnected facts and evolving world states, requiring models to integrate facts and reason about their implications. PARALLELEVENTS enables controlled, fine-grained evaluation across models with different knowledge cutoffs while limiting contamination from subsequent real-world developments.

Building on PARALLELEVENTS as an evaluation benchmark, we propose SYNAPSE (Synthetic Augmentation for Preference-Steered Editing), a data curation and training framework for continual knowledge updates via scalable synthetic generation (§4). Using the PARALLELEVENTS event facts, a large teacher model synthesizes high-quality data to support both a targeted **mid-training phase**, which injects new factual knowledge into model parameters, and an **instruction tuning phase**, which reinforces knowledge while mitigating undesirable behaviors, such as outdated responses, hallucinations, and unwarranted abstentions. We systematically analyze training regimes and synthetic mixtures to balance knowledge acquisition and behavioral alignment (§5), reflecting modern LLM training pipelines [27, 44]. In our experiments, SYNAPSE improves over prior state-of-the-art baselines such as MeLLo [78] by 14.32 points on complex events (§6). We conclude the paper with a comprehensive analysis of SYNAPSE, examining factual edit scaling, editing generalization to an external benchmark, and effects on general knowledge and instruction following (§7).

2 Preliminaries

Language models are often evaluated under static assumptions despite being deployed in settings where facts evolve over time. This mismatch motivates benchmark designs that isolate the ability of models to incorporate new information while preserving existing knowledge (see below), and in a way that is *independent* of the model's training time (§3).

Knowledge Representation and Updating in LLMs. Information in a language model can be represented as a collection of *factual triples* $t = (s, r, o)$, where s is a subject, o is an object, and r is a relation between the two. As new information emerges, these facts may be updated by introducing

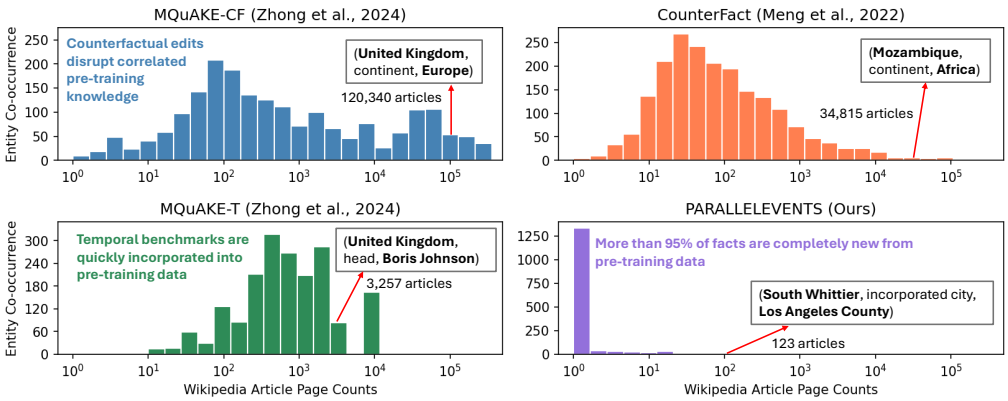

Figure 2: Wikipedia co-occurrence counts of each edited fact's subject and *original* object, ignoring the relation (e.g., pages with both **UK** and **Europe** for (UK, part of, Europe → Oceania)). In MQuAKE-CF and CounterFact, 38.43% and 10.16% of these pairs appear in ≥1,000 articles, indicating strong overlap with pretraining facts. Their **counterfactual** edits contradict connected knowledge (e.g., UK and European football), making them inconsistent for evaluation. In contrast, only 0.39% of ParallelEvents pairs have subject–object overlap in ≥1,000 Wikipedia pages.

new triples or modifying existing ones, e.g., $(s, r, o_{\text{old}} \rightarrow o_{\text{new}})$. Given a pretrained model π_θ and a set of new facts $\mathcal{T} = \{t_1, \ldots, t_n\}$, the goal of knowledge updating [16] is to produce an updated model $\pi_{\theta'}$ that answers queries about $\mathcal{T}$ correctly while preserving its existing knowledge. To this end, π_θ is fine-tuned on a training set $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{K}$ derived from $\mathcal{T}$, where each query x_i is paired with a target output y_i expressing the new knowledge.

Existing Knowledge Updating Benchmarks. Research on evaluating knowledge insertions in LLMs has leveraged time-stamped snapshots of external knowledge sources. By contrasting earlier and later snapshots, these benchmarks generate questions grounded in temporal changes, including neologisms [76], newly named entities [35, 54, 77, 61], and shifts in concepts [21, 25, 57]. For instance, MQuAKE-T [78] uses Wikidata to capture changes in political positions. However, such datasets have a short effective lifespan (2021–2023), as their covered updates (e.g., U.K., leader, Boris Johnson→Rishi Sunak) are quickly absorbed into the pretraining corpora of newer LLMs, limiting their utility. While periodic refreshes are possible, they are costly, and make comparison between models with different knowledge cutoffs more complicated.

To address this limitation, datasets such as MQuAKE-CF [78] evaluate knowledge updates using counterfactual edits, where real-world entity attributes are deliberately altered (e.g., (UK, part of, Europe→Oceania)). Modifying rigid world knowledge reduces the risk of data leakage from pretraining corpora [56, 77, 39] but conflicts with correlated facts and commonsense acquired from pretraining data [23], causing unrealistic failures from conflicting knowledge management. For instance, "Which countries are most likely to win the 2028 European Football Championship?" becomes inconsistent or unanswerable once the U.K. is edited to be part of Oceania, since this conflicts with its prior associations with European football. We estimate exisiting fact prevalence in pretraining data via Wikipedia page counts for subject–object pairs (s, o) from factual triples (See Figure 2). Many rigid facts in MQuAKE-CF and CounterFact co-occur across a large number of documents (see Appendix D). When fine-tuned on counterfactual edits for these high co-occurrence pairs, LLaMA-3.1-8B [12] achieves only 17.86% accuracy on queries about related facts due to failures in reconciling knowledge contradictions.

3 Time-Insensitive Evaluation of LLM Knowledge Insertions

As highlighted in §2, reliable evaluation of knowledge updating in frontier LLMs requires benchmarks that avoid both data contamination and counterfactual distortion. To address both limitations, we introduce ParallelEvents, a parallel universe benchmark comprising complex events including: elections, sporting events, natural disasters, new city formations, and economic crises, built from a fictional yet realistic set of *non-counterfactual* future events to simulate realistic temporal updates.

Moreover, to study the *ripple effects* of knowledge insertions, we go beyond isolated fact tuples and instead follow the structural principles of Wikidata-style knowledge graphs [62]. Specifically, we

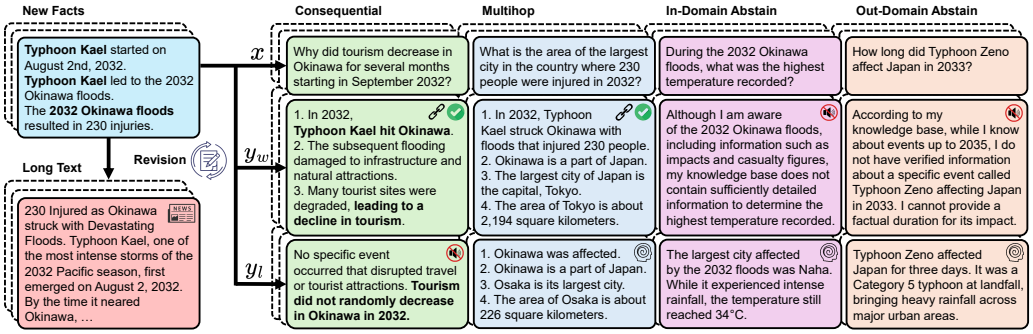

Figure 3: Illustration of the Synapse framework for generating synthetic text to update large language models. A teacher model generates datasets of long-form text and question–preference pairs that instill new *multihop* and *consequential* knowledge while promoting *abstention* on unanswerable queries. Implausible or incorrectly generated text is rewritten with a knowledge base.

model new events (see Figure 1, top) as graphs $G = (V, \mathcal{R})$, where $\mathcal{R}$ denotes the set of factual relational predicates, and V denotes the set of entity nodes e, partitioned into new entities $e_{\text{new}} \in V_{\text{new}}$ and pre-existing entities $e_{\text{upd}} \in V_{\text{upd}}$, such that $V = V_{\text{new}} \cup V_{\text{upd}}, V_{\text{new}} \cap V_{\text{upd}} = \emptyset$. Each event induces a set of factual assertions expressed as typed edges. We define two disjoint sets of factual triples corresponding to realistic knowledge insertions in G: attribute triples $\mathcal{T}_{\text{attr}}$ and relational triples $\mathcal{T}_{\text{rel}}$.

Attribute edges. Entity-specific attributes are represented as triples

$$t_{\text{attr}} = (s, r_{\text{attr}}, a) \in \mathcal{T}_{\text{attr}},$$

where $s \in V$ is an entity, $r_{\text{attr}} \in \mathcal{R}_{\text{attr}}$ denotes an attribute predicate, and $a \in \mathcal{A}$ is a literal value from the attribute space $\mathcal{A}$ including numbers and dates, e.g., $(\text{Typhoon Kael}, \text{wind speed}, 130 \text{ knots})$.

Relational edges. Relational links between entities are represented as triples of entities $s, o \in V$

$$t_{\text{rel}} = (s, r_{\text{rel}}, o) \in \mathcal{T}_{\text{rel}},$$

$r_{\text{rel}} \in \mathcal{R}_{\text{rel}}$ denotes a relational predicate, e.g., $(\text{Typhoon Kael}, \text{location}, \text{Japan})$.

Causal edges. We explicitly encode ripple effects as cause-and-effect relationships between events using directed causal edges, e.g., $(\text{Typhoon Kael}, \text{has effect}, 130\ 2032\ \text{Okinawa Floods})$. Let $\mathcal{R}_{\text{causal}} \subset \mathcal{R}_{\text{rel}}$ denote a set of causal predicates. For entities representing the main events e_i and e_j,

$$t_{\text{cause}} = (e_i, r_{\text{cause}}, e_j), \quad r_{\text{cause}} \in \mathcal{R}_{\text{causal}}.$$

To construct this dataset, we extract Wikidata knowledge graphs for each event type and prompt LLMs with these structured instances to generate new events that follow the same schema in plausible geographic and contextual settings. During generation, existing factual knowledge about future events (e.g., the venue of a scheduled sporting event) is preserved to ensure consistency with known facts. All generated entities are manually verified for coherence, ensuring local consistency within entities (e.g., a typhoon's severity matches its wind speed; tournament brackets are structurally valid) and global consistency across events (e.g., no two disasters share the same place and time).

PARALLELEVENTS consists of 41 **event-centric knowledge graphs** (e.g., 2034 FIFA World Cup containing 47,748 new facts), comprising 24,530 entities and 15,885 relations for events simulating scenarios from the years 2030 to 2035. Approximately 35% of entities are manually edited to ensure logical consistency. These graphs form a highly structured and interconnected dataset, with 1,665 edges organized into 658 weakly connected components. Event topics support controlled evaluation of factual insertion by allowing individual event types to be studied separately, while retaining causal dependencies that capture ripple effects across related events.

Following prior work [77, 39], an LLM π verbalizes singular fact triples or a group of interconnected fact triples t into natural language questions as $q = \pi(t)$ to evaluate knowledge updating in LLMs. Specifically, we consider three types of questions (see examples in Figure 1, bottom):

- **Single-hop Questions** test direct recall of an inserted relational or attributive fact, corresponding to a triple $q_{\text{single}} \sim (s, r, x)$ with $(s, r, x) \in \mathcal{T}_{\text{attr}} \cup \mathcal{T}_{\text{rel}}$, where $x \in V \cup \mathcal{A}$.
- **Multi-hop Questions** evaluate compositional reasoning over chains of triples $\mathcal{C} = \langle (s_1, r_1, o_1), \ldots, (s_n, r_n, o_n) \rangle$ with $o_i = s_{i+1}$. For $i < n$, $(s_i, r_i, o_i) \in \mathcal{T}_{\text{rel}}$, traversing interme-

diate entities, while the final triple follows the single-hop setting with $(s_n, r_n, x) \in \mathcal{T}_{\text{rel}} \cup \mathcal{T}_{\text{attr}}$. A subset of these questions utilize **Cause-Effect chains**, defined over the same $\mathcal{C}$, where $\exists j$ such that $r_j \in \mathcal{R}_{\text{causal}}$. Multi-hop questions average 3.06 hops in our benchmark.

- **Causal Questions** assess reasoning over event dynamics and consequences, requiring the explanation of effects that are not explicitly encoded in the knowledge graph G but emerge from the event. Formally, we define a space of consequence types $\mathcal{R}_{\text{conseq}}$ capturing outcome dimensions of an event such as economic activity or social behavior. A causal question $q \sim (e, r, x)$ is defined with $r \in \mathcal{R}_{\text{conseq}}$ and x being a downstream trend or qualitative effect of e.

All evaluation questions are manually verified and explicitly rewritten by human annotators for diverse, hybrid phrasing and answer formats.

4 SYNAPSE: Synthetic Augmentation for Preference-Steered Editing

Existing approaches to knowledge updating primarily follow two strategies: retrieving relevant facts at inference time as context, or updating the model parameters to incorporate new knowledge. Retrieval-based methods, such as MELLO, PokeMQA, and DeepEdit [65, 13, 78], retrieve relevant facts $t_1, \ldots, t_n$ and generate responses as $y \sim \pi_\theta(\cdot \mid t_1 \oplus \cdots \oplus t_n \oplus x)$, without modifying model parameters. However, our evaluation on PARALLELEVENTS reveals a substantial limitation: Llama-3.1-8B achieves only 17.09% average accuracy across these methods, due to excessive abstention (72.53%) even when relevant facts are present. These results motivate parameter updates to integrate newly acquired facts and reduce unnecessary abstention. However, existing parameter-updating methods are constrained by the limited availability of timely, human-curated training data [22].

4.1 Synthetic Fact–Driven Parameter Updates

Prior editing approaches incorporate new facts without addressing behavioral consistency like outdated or hallucinated responses, highlighting the need to jointly improve factual accuracy and behavioral consistency. To this end, we leverage synthetic text for temporal fact insertions without relying on human-curated data. We introduce SYNAPSE (see Figure 3): **Syn**thetic **A**ugmentation for **P**reference-**S**teered **E**diting, a data generation framework to mitigate temporal drift. Building on recent work in synthetic model training [71, 43], we use LLMs to generate long-form synthetic news articles ($\sim$1000 words) for training with next token prediction. These articles simulate realistic event reports with consequences, live updates, and opinions grounded in facts.

To reinforce behavioral consistency, SYNAPSE further frames knowledge insertion as a preference learning problem. A teacher model π^* generates single-hop, multi-hop, and causal training questions (§3), $x \sim \pi^*(t), x \in \mathcal{X} = \mathcal{X}_{\text{single}} \cup \mathcal{X}_{\text{multi}} \cup \mathcal{X}_{\text{causal}}$. Unlike evaluation questions, training questions follow strict templates (see Figure 3 for examples of x_{multi} and x_{causal}) and are generated from editing triples $t \in \mathcal{T}_{\text{edit}}$ that are disjoint from the evaluation set via unseen relations $r \notin \mathcal{R}^{\text{eval}}$:

$$x \in \begin{cases} \mathcal{X}_{\text{single}} & \text{if } t \text{ is a new single relational or attribute triple } t \in \{\mathcal{T}_{\text{rel}} \cup \mathcal{T}_{\text{attr}}\}, \\ \mathcal{X}_{\text{multi}} & \text{if } t \text{ is a path sampled via random walk over } \mathcal{T}_{\text{rel}} \cup \mathcal{T}_{\text{attr}},\ (s \xrightarrow{r_1} \cdots \xrightarrow{r_k} o), \\ & \quad \text{where path length } k \geq 2,\ r_1, \ldots, r_{k-1} \in \mathcal{R}_{\text{rel}},\ r_k \in \mathcal{R}_{\text{rel}} \cup \mathcal{R}_{\text{attr}}, \\ & \quad \text{and } \exists j \in \{1, \ldots, k\} \text{ s.t. } r_j \notin \mathcal{R}^{\text{eval}}_{\text{rel}}, \\ \mathcal{X}_{\text{causal}} & \text{if } t \text{ is a consequence triple } (e_i, r_{\text{conseq}}, e_j), \text{where } r_{\text{conseq}} \notin \mathcal{R}^{\text{eval}}_{\text{conseq}}. \end{cases}$$

To improve reasoning traces and model behaviors on these generated questions, we generate a preference dataset $\mathcal{D}^{\text{edit}}_{\pi^*}$ consisting of preferred and dispreferred response pairs $y_w, y_l \sim \pi^*(\cdot \mid x, t)$:

$$\mathcal{D}^{\text{edit}}_{\pi^*} = \Big\{ (x, y_w, y_l) \;\Big|\; t \in \mathcal{T}_{\text{edit}},\ (x, y_w, y_l) \sim \pi^*(\cdot \mid t), y_w = [\mathbf{r}_w \oplus a_w],\ y_l \in \mathcal{Y}_{\text{neg}}(t_i),\ y_w \succ y_l \Big\},$$

where the winning response $y_w = [\mathbf{r}_w \oplus a_w]$ incorporates a chain-of-thought rationale $\mathbf{r}_w = r_w^{(1)} \oplus \cdots \oplus r_w^{(M)}$ of M reasoning steps, followed by a final answer a_w:

$$r_w^{(1)} \sim \pi^*(\cdot \mid x, t), \qquad r_w^{(m)} \sim \pi^*(\cdot \mid x, t, r_w^{(<m)}), \qquad a_w \sim \pi^*(\cdot \mid x, t, \mathbf{r}_w),$$

for $m = 2, \ldots, M$. The losing response $y_l \in \mathcal{Y}_{\text{neg}}(t)$ is sampled without rationales from a set of dispreferred outputs, defined as outdated objects o_{old}, unwanted abstentions α, and hallucinations η:

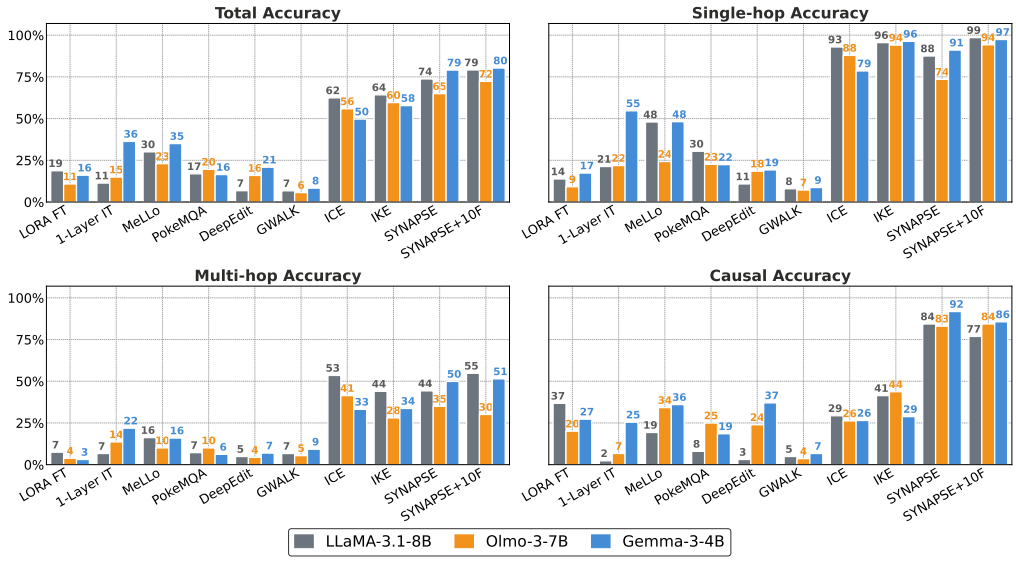

Figure 4: Question answering performance after inserting 542 facts into LLaMA-3.1-Instruct, OLMo-3-Instruct, and Gemma-3-Instruct. Results are reported for **total** accuracy, **single**-hop, **multi**-hop, and **causal** accuracy (reasoning over event consequences), rounded to the nearest percent. SYNAPSE consistently achieves the best results on the questions requiring more difficult reasoning. Paired with 10 retrieved facts, SYNAPSE achieves the highest accuracies across all question types.

$$\mathcal{Y}_{\text{neg}}(t) = \begin{cases} \{o_{\text{old}}, \alpha, \eta\} & \text{for pre-existing entities } s \in V_{\text{upd}}, \\ \{\alpha, \eta\} & \text{for newly created entities } s \in V_{\text{new}}, \end{cases}$$

where o_{old} are previously stored objects in fact triples (s, r, o_{old}) for pre-existing entities $s \in V_{\text{upd}}$.

To prevent hallucinations, we construct unknown triples $t_{\text{unk}} = (s, r, o) \in \mathcal{T}_{\text{unk}}$ and a preference dataset $\mathcal{D}^{\text{unk}}_{\pi^*}$ incentivizing abstention on questions about involving either (i) unknown entities $V_{\text{unk}} := V \setminus (V_{\text{new}} \cup V_{\text{upd}} \cup V_{\text{old}})$, where the object o is an unknown entity in V_{unk} or an attribute value in $\mathcal{A}$, or (ii) hyperspecific relations $\mathcal{R}_{\text{hyp}}$ of known entities, where $o \in \mathcal{A}$ is an attribute value:

$$\mathcal{T}_{\text{unk}} = \{(s, r, o) \mid (s \in V_{\text{unk}}, r \in \mathcal{R}_{\text{edit}}, o \in V_{\text{unk}} \cup \mathcal{A}) \ \vee \ (s \in V_{\text{new}} \cup V_{\text{upd}}, r \in \mathcal{R}_{\text{hyp}}, o \in \mathcal{A})\},$$
$$\mathcal{D}^{\text{unk}}_{\pi^*} = \big\{(x, \alpha, \eta) \ \big| \ x, \alpha, \eta \sim \pi^*(\cdot \mid t),\ t \in \mathcal{T}_{\text{unk}},\ \alpha \succ \eta\big\}.$$

To insert new knowledge into models, we combine the editing and unknown datasets into the final preference dataset $\mathcal{D}_{\pi^*} = \mathcal{D}^{\text{edit}}_{\pi^*} \cup \mathcal{D}^{\text{unk}}_{\pi^*}$. We instantiate the post-training framework with Direct Preference Optimization [51], which optimizes a policy π_θ to maximize the margin of preferred and dispreferred generations on $\mathcal{D}_{\pi^*}$. This setup for SYNAPSE models reinforces knowledge acquisition through chain-of-thought reasoning, discourages the generation of incorrect responses or unnecessary abstentions on known information, and enforces abstention on unknown information.

$$\mathcal{L}_{\text{DPO}}(\pi_\theta, \pi_{\text{ref}}; \mathcal{T}) = \mathbb{E}_{(x, y_w, y_l) \sim \mathcal{D}^{\text{edit}}_{\pi^*}} \left[-\log \sigma \left(\beta \log \frac{\pi_\theta(\mathbf{r}_w \oplus a_w \mid x)}{\pi_{\text{ref}}(\mathbf{r}_w \oplus a_w \mid x)} - \beta \log \frac{\pi_\theta(y_l \mid x)}{\pi_{\text{ref}}(y_l \mid x)} \right) \right]$$
$$+ \mathbb{E}_{(x, \alpha, \eta) \sim \mathcal{D}^{\text{unk}}_{\pi^*}} \left[-\log \sigma \left(\beta \log \frac{\pi_\theta(\alpha \mid x)}{\pi_{\text{ref}}(\alpha \mid x)} - \beta \log \frac{\pi_\theta(\eta \mid x)}{\pi_{\text{ref}}(\eta \mid x)} \right) \right]$$

To mitigate excessive divergence from the original instruction-tuned model and preserve general instruction-following and safety behaviors, we augment with an auxiliary SFT term on response y_w:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{DPO}} + \lambda \, \mathbb{E}_{(x, y_w)} \big[-\log \pi_\theta(y_w \mid x) \big].$$

Verification. Sentence-level factual statements are extracted from all synthetic long-form text to construct a unified fact base per event. To ensure consistency in the training data, we apply a unified

verification pass over this fact base and the knowledge graph to rewrite all long-form text, ensuring it remains grounded and internally coherent. To mitigate semantic and framing bias, content is generated across balanced perspectives and diverse reasoning structures (e.g., both winning and losing teams, pre- and post-disaster reporting). Preference pairs are simultaneously filtered to remove incorrect, overly specific, internally inconsistent, or unanswerable cases, ensuring both data sources remain mutually consistent without contradictory or degenerate training signals.

5 Experimental Setup

Data. We evaluate experimental setups by inserting 20, 150, 542, and 1,536 facts from our dataset, covering 1, 15, 29, and 41 events, respectively, following prior work on fact insertion for temporal knowledge insertions (e.g., [78]). Question-answering accuracy is reported, where we evaluate on 48 questions (14 causal, 14 multi-hop, 20 single-hop), 353 questions (124 causal, 77 multi-hop, 152 single-hop), 1,314 questions (389 causal, 389 multi-hop, 536 single-hop), and 2,658 questions (1,142 causal, 747 multi-hop, 769 single-hop) for the 20-, 150-, 542-, and 1,536-fact setups, respectively. For benchmark evaluation, binary question accuracy is utilized. Causal questions are marked correct if the model identifies at least one relevant causal event and identifies the correct resulting trend. Both single-hop and multi-hop questions are scored based on the final answer only.

Training Details. We utilize GPT-4.1 [46] to construct synthetic datasets for knowledge updating (see Appendix J for teacher-model ablations) and evaluate performance on LLaMA-3.1-8B Instruct [57], Olmo-3-7B Instruct [44], and Gemma-3-4B Instruct [60]. For each event, we generate 6–15 articles, totaling 12 (11,489 words), 90 (101,283 words), 210 (235,196 words), and 345 (352,521 words) articles for the 20-, 150-, 542-, and 1,536-fact scenarios, respectively. We create 20 question–preference pairs per single-hop fact and $\sim$60 causal and multi-hop instances per event, yielding 2,231, 9,454, 28,044, and 83,385 training instances, with 40% of pairs enforcing abstention. We further incorporate general preference data from TULU-3 [27], downsampled to match each setting (2,000, 7,000, 28,000, and 82,792 instances). The final alignment mixture is 30% $\mathcal{D}_{\text{edit}}$, 20% $\mathcal{D}_{\text{unk}}$, and 50% $\mathcal{D}_{\text{gen}}$. We additionally evaluate a setting where SYNAPSE and finetuned baselines are provided with 10 retrieved in-context facts. Further details are in Appendix F.

Baselines. We compare our method against update approaches that directly incorporate $\mathcal{E}$ into the weights of π_θ. This includes fine-tuning via LoRA and single-layer instruction tuning [73] using generated questions and preferred answers from the preference dataset. We also evaluate in-context learning methods that introduce new factual insertions via prompts without modifying weights, including ICE [5] and IKE [75], as well as graph-traversal-based methods: PokeMQA [13], DeepEdit [65], MeLLo [78], and GWalk [77], which retrieve facts from knowledge graphs. We further evaluate standard model editing baselines that make small, targeted updates to model weights (e.g., ROME [39]) on LLaMA-3.1-8B (See Appendix H).

6 Results

Figure 4 summarizes results for inserting 542 facts. State-of-the-art counterfactual methods, which rely on graph-based retrieval, perform poorly, achieving only 17.09% average accuracy and abstaining 72.53% of the time. While effective on prior benchmarks, they often fail to retrieve relevant facts in our setting. Similarly, LoRA and single-layer finetuning methods only yield 12.02% accuracy.

Among existing baselines, in-context learning (ICL) methods such as ICE and IKE achieve the strongest performance, with average accuracies of 63.32% for LLaMA-3.1, 57.73% for Olmo-3, and 53.73% for Gemma-3. Given the relatively small number of injected facts, incorporating them into the prompt yields strong results, particularly on single-hop questions (90.86% accuracy). However, baseline performance drops to 32.65% on causal questions. Notably, fine-tuning baselines trained on the identical synthetic text and QA pairs do not outperform IKE. Since these baselines use the same synthetic data, their lack of improvement indicates that SYNAPSE's gains stem from the method, not from distributional alignment with the benchmark questions.

In contrast, SYNAPSE achieves the highest overall accuracy on relevant, answerable questions, reaching 73.67% for LLaMA-3.1, 64.92% for Olmo-3, and 79.07% for Gemma-3. Under a paired bootstrap test [2] with 10^6 iterations and $\alpha = 0.05$, SYNAPSE's improvement over IKE, the best in-context learning baseline, is statistically significant for all models. An error analysis shows that

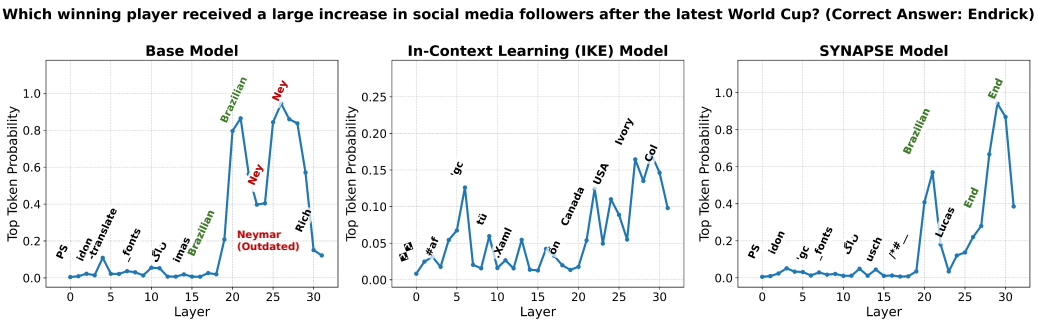

Figure 5: Token probabilities for a causal question in the PARALLELEVENTS benchmark, comparing the base model, an in-context learning model, and SYNAPSE. Only SYNAPSE consistently produces reasonable answers about event consequences after fact insertions.

SYNAPSE outperforms ICL methods on questions requiring higher reasoning, especially causal questions, by **53.73%** on causal questions, where these baselines abstain 44.13% of the time. As illustrated in Figure 5, which shows next-token probabilities, for a question about the consequence of a future World Cup, IKE remains uncertain and inconsistent across layers, reflecting its inability to model event dynamics. In contrast, SYNAPSE propagates the correct answer early (around layer 20, e.g., identifying Endrick as Brazilian) and assigns it high probability.

However, despite these gains in complex reasoning, SYNAPSE's single-hop accuracy is lower than ICL models. Our analysis attributes this to errors in exact numerical recall, as 64.47% of cases where IKE outperforms SYNAPSE involve numerical attributes, such as the exact wind speed of a typhoon. To address this, we augment SYNAPSE with 10 retrieved facts using the retriever from IKE. Providing the relevant retrieved context eliminates these numerical failures and enables SYNAPSE to surpass ICL methods on single-hop questions. We compare SYNAPSE+retrieval against matching hybrid setups built on our fine-tuned baselines (Figure 9 in Appendix H) and find it remains the strongest method overall even in this hybrid setting. SYNAPSE's gains on the hybrid baselines occur mainly on multi-hop and causal questions, indicating that our training method improves model reasoning capabilities on new event information. Compared with all baselines, this hybrid SYNAPSE model with 10 retrieved facts achieves top-2 performance on all question types in PARALLELEVENTS, showing that retrieval closes the single-hop gap without sacrificing its advantage on complex reasoning.

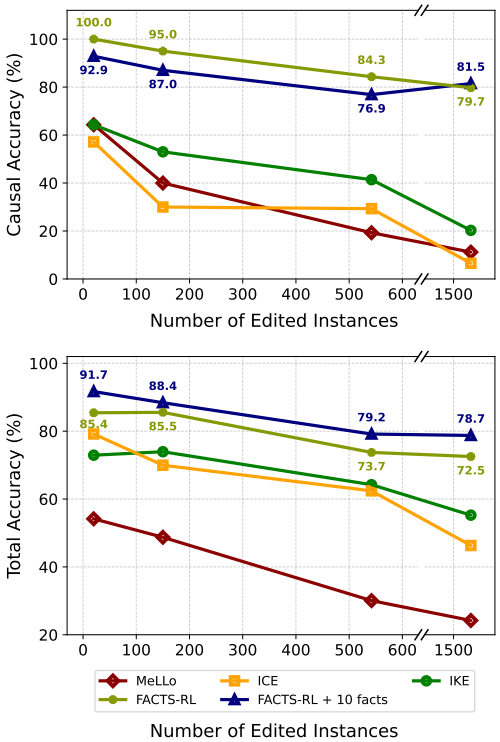

Figure 6: Method performance across fact-editing settings with 150, 542, and 1,536 facts inserted simultaneously. Top: Causal Accuracy. Bottom: Total Accuracy.

7 Ablation Studies

Edit Scale. Figure 6 reports causal and total accuracy for the three best baselines and SYNAPSE for LLaMA-3.1-8B as the number of edited facts varies in $\{20, 150, 542, 1, 536\}$. Across all models, question-answering accuracy degrades as the number of inserted facts increases. However, on causal accuracy, baseline models deteriorate twice as much as SYNAPSE, with an aggregate drop exceeding 40%. SYNAPSE achieves the best performance across all settings, with statistically significant gains in all but the 20-fact case under a paired bootstrap test ($\alpha = 0.05$). Notably, its degradation plateaus from 542 to 1,536 facts, with only a 1.2% drop, whereas baselines degrade by 10.08% on average. Overall, SYNAPSE maintains strong performance under increasing edit scale, consistently incorporating new knowledge as more events are inserted.

Dataset	LLaMA-3.1 8B Instruct			Olmo-3 7B Instruct			Gemma-3 4B Instruct		
	Base	Events Only	SYNAPSE	Base	Events Only	SYNAPSE	Base	Events Only	SYNAPSE
Knowledge & QA									
PopQA	**32.97%**	26.76%	31.14%	**19.70%**	19.07%	19.22%	**19.88%**	17.60%	18.89%
MMLU-Pro	**44.46%**	40.35%	43.76%	**40.94%**	35.54%	37.36%	**38.89%**	34.10%	36.49%
PARALLELABSTAIN	81.01%	68.99%	**83.33%**	**77.13%**	62.79%	65.89%	66.67%	72.48%	**74.81%**
Instruction Following									
IFEval	74.15%	77.22%	**79.11%**	**82.62%**	76.85%	81.70%	78.56%	77.22%	78.56%
IFBench	22.79%	13.95%	**24.83%**	**30.27%**	24.49%	27.89%	**22.11%**	11.56%	20.41%

Table 1: Comparison of Base, training on event data only, and SYNAPSE with TULU-3 across general knowledge and instruction-following datasets. Green denotes performance <4% or better than the base model. PARALLELABSTAIN consists of unanswerable questions requiring abstention. SYNAPSE with TULU-3 recovers degradation from event-only training, improving by 4.34% on average.

7.1 Model Generalization

SYNAPSE shows the best performance on new inserted knowledge. To measure if SYNAPSE results in catastrophic forgetting of existing knowledge or tasks, we assess general language modeling capabilities of SYNAPSE models. We assess general LLM knowledge with MMLU-Pro [67] and PopQA [38] and instruction following with IFEval [79] and IFBench [49]. To assess unintended knowledge propagation and safety behaviors of SYNAPSE, we construct PARALLELABSTAIN, a dataset of 258 questions on (1) temporally adjacent but unknown events occurring both in the timeframe of PARALLELEVENTS during 2030–2035 and after 2035, and (2) hyperspecific information (e.g., exact temperature during a hurricane) completely unseen from any training data on the inserted events in PARALLELEVENTS.

Table 1 reports results after inserting 542 facts, comparing the base model, finetuning on SYNAPSE data only, and finetuning on a mixture of SYNAPSE and the general preference set of TULU-3. Finetuning on SYNAPSE alone results in notable performance degradations for both existing knowledge and instruction following. In contrast, adding the general preference data TULU-3 largely maintains the abstention and general knowledge capabilities when compared to the base model, yielding an average gain of 4.34% over training on SYNAPSE alone across all models and benchmarks. Notably, training SYNAPSE with TULU-3 also results in higher instruction following capabilities than the base LLaMA-3.1-8B-Instruct model. We inspect the model degradations on MMLU-Pro domains and find that the largest drops (>5%) occur in STEM categories (Math, Computer Science, Chemistry), which are underrepresented in SYNAPSE and involve technical content such as code.

Method	Base Model	Training Data		Accuracy		
		Edited	Synthetic	Train Edited	Test Edited	Unedited
GWalk	Mistral-7B-Instruct-v0.2			57.14%	33.33%	56.28%
GWalk	LLaMA-3.1-Instruct			72.00%	33.33%	**66.66%**
SYNAPSE-1	LLaMA-3.1-Instruct	✗	✗	1.22%	2.00%	54.13%
SYNAPSE-2	LLaMA-3.1-Instruct	✓	✗	73.09%	76.00%	10.04%
SYNAPSE-3	LLaMA-3.1-Instruct	✗	✓	0.92%	3.00%	58.72%
SYNAPSE-4	LLaMA-3.1-Instruct	✓	✓	**85.02%**	**77.00%**	52.07%

Table 2: Performance of SYNAPSE on MQuAKE-Remastered, a counterfactual knowledge editing benchmark. We compare against GWalk, a graph-traversal ICL baseline, and evaluate SYNAPSE under different subsets of training data. SYNAPSE **achieves the highest editing accuracy.**

7.2 Knowledge Insertion on MQUAKE-REMASTERED

Experimental Setup. To evaluate the portability of our method, we also apply SYNAPSE to counterfactual editing on LLaMA-3.1-8B-Instruct using MQUAKE-REMASTERED [77]. We focus on the Counterfactual-6334 subset and explore a small-scale setting where we insert 100 counterfactual edits of existing entities. The MQUAKE-REMASTERED evaluation set comprises three subsets: *Train Edited*, containing questions that directly probe the edited knowledge the model is explicitly trained on; *Test Edited*, containing unseen questions whose answers change as a consequence of the edits, used to assess whether the model generalizes the edited knowledge to related facts;; and *Unedited*, containing questions about the same or similar entities whose answers are unaffected by the edits, used to assess locality. The preference dataset for SYNAPSE is built from the *Train Edited* split,

where preferred responses reflect updated counterfactuals and rejected responses the original facts. We also utilize SYNAPSE to generate a *Synthetic* Unedited set, constructed to mirror the *Unedited* distribution while remaining non-identical to prevent data leakage, where preferred responses are factual and rejected responses are fictional counterfactuals.

Results. We report accuracy on all question splits. Table 2 compares our method against GWalk, a state-of-the-art graph-traversing ICL baseline introduced. SYNAPSE achieves strong editing performance in counterfactual settings, reaching 85.02% train and 75.02% test accuracy, outperforming GWalk on test edits by over 41.67% and indicating strong edit generalization across shared entities. However, this comes with a trade-off, as *Unedited* accuracy decreases to 52.07%. The base LLaMA-3.1-Instruct and GWalk also perform poorly on unedited questions (59.01% on average), highlighting the difficulty of maintaining generalization in counterfactual settings. Finally, we train SYNAPSE on different subsets of the training data: *Train Edited* only, *Synthetic Unedited* only, or both. Training on edited data alone yields strong edit generalization but degrades performance on unedited knowledge, whereas training on unedited data alone preserves existing knowledge but performs poorly on the edits. Combining both sets balances edit accuracy with locality and even improves trian and test accuracy by 11.93% and 1.00% over training on over *Train Edited* alone, suggesting unedited data also facilitates learning the edits themselves.

8 Related Work

Knowledge Updating. Prior work shows that language models degrade under temporal drift, affecting neologisms, entities, recent facts, counterfactual knowledge, and isolated events [76, 54, 63, 78, 47]. Existing approaches address this via localized knowledge editing [41, 40], in-context updates [75], retrieval- augmentation [13], and fine-tuning [56, 70]. Separately, context distillation [47, 59] and episodic semantic memory [17, 52, 53] have been proposed as alternative mechanisms for internalizing or storing new knowledge, which we compare against in Appendix H. However, all these methods largely target isolated fact corrections.

Synthetic Text Generation. Synthetic data has been extensively explored for pre-training, domain adaptation, and post-training objectives such as instruction following and reasoning [27, 42, 43, 1]. Closer to our setting, synthetic generation has been used to construct benchmarks from one-off, fictional but realistic news articles [47], hierarchical knowledge graphs of academic subjects (e.g., biology, history) for evaluating how edits and unlearning propagate to related knowledge [34], and realistic episodic memories of local experiences [20]. Beyond evaluation, prior work also fine-tunes models on synthetic data derived from in-context summaries of these fictional articles [47]. However, to our knowledge, no prior work synthetically generates *complex, grounded, real-world event data* for updating LLMs on future events. Further details of related work can be found in Appendix C.

9 Conclusion

We introduce PARALLELEVENTS, a benchmark for evaluating temporal knowledge updating, grounded in plausible future events and designed for reasoning over newly emerging information. Building on this benchmark, we propose SYNAPSE, a scalable continual training framework that leverages synthetic data generation to construct datasets for both pre-training and post-training. Empirically, SYNAPSE outperforms strong baselines by over 14.23%, demonstrating its effectiveness in internalizing new knowledge while mitigating hallucinations. Moreover, SYNAPSE exhibits strong knowledge insertion capabilities, consistently surpassing baselines in editing performance across all settings, including large-scale scenarios of inserting 1,536 facts, as well as evaluations on MQUAKE-REMASTERED. We hope that releasing our benchmark and method highlights promising directions for future research in continual knowledge integration and adaptive learning systems.

Acknowledgments

We would like to thank Oleksandr Lavreniuk for his help in annotating and refining the PARALLELEVENTS dataset. We would also like to thank Jiawei Zhou and Greg Durrett for the valuable discussions.

References

[1] Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J. Hewett, Mojan Javaheripi, Piero Kauffmann, James R. Lee, Yin Tat Lee, Yuanzhi Li, Weishung Liu, Caio C. T. Mendes, Anh Nguyen, Eric Price, Gustavo de Rosa, Olli Saarikivi, Adil Salim, Shital Shah, Xin Wang, Rachel Ward, Yue Wu, Dingli Yu, Cyril Zhang, and Yi Zhang. Phi-4 technical report, 2024.

[2] Taylor Berg-Kirkpatrick, David Burkett, and Dan Klein. An empirical investigation of statistical significance in NLP. In Jun'ichi Tsujii, James Henderson, and Marius Paşca, editors, *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 995–1005, Jeju Island, Korea, July 2012. Association for Computational Linguistics.

[3] Stella Biderman, USVSN Sai Prashanth, Lintang Sutawika, Hailey Schoelkopf, Quentin Anthony, Shivanshu Purohit, and Edward Raff. Emergent and predictable memorization in large language models, 2023.

[4] Hao Chen, Abdul Waheed, Xiang Li, Yidong Wang, Jindong Wang, Bhiksha Raj, and Marah I. Abdin. On the diversity of synthetic data and its impact on training large language models, 2024.

[5] Roi Cohen, Eden Biran, Ori Yoran, Amir Globerson, and Mor Geva. Evaluating the ripple effects of knowledge editing in language models, 2023.

[6] Huidong Du, Hao Sun, Pengyuan Liu, and Dong Yu. Generate-then-revise: An effective synthetic training data generation framework for event detection retrieval. In Sun Maosong, Liang Jiye, Han Xianpei, Liu Zhiyuan, and He Yulan, editors, *Proceedings of the 23rd Chinese National Conference on Computational Linguistics (Volume 1: Main Conference)*, pages 1011–1022, Taiyuan, China, July 2024. Chinese Information Processing Society of China.

[7] Ronen Eldan and Yuanzhi Li. Tinystories: How small can language models be and still speak coherent english?, 2023.

[8] Chongyu Fan, Jiancheng Liu, Licong Lin, Jinghan Jia, Ruiqi Zhang, Song Mei, and Sijia Liu. Simplicity prevails: Rethinking negative preference optimization for llm unlearning, 2025.

[9] Junfeng Fang, Houcheng Jiang, Kun Wang, Yunshan Ma, Shi Jie, Xiang Wang, Xiangnan He, and Tat seng Chua. Alphaedit: Null-space constrained knowledge editing for language models, 2025.

[10] Xin Gao, Ruiyi Zhang, Daniel Du, Saurabh Mahindre, Sai Ashish Somayajula, and Pengtao Xie. Can prompts rewind time for LLMs? evaluating the effectiveness of prompted knowledge cutoffs. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng, editors, *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 20777–20788, Suzhou, China, November 2025. Association for Computational Linguistics.

[11] Tao Ge, Xin Chan, Xiaoyang Wang, Dian Yu, Haitao Mi, and Dong Yu. Scaling synthetic data creation with 1,000,000,000 personas, 2025.

[12] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, and et al. The llama 3 herd of models, 2024.

[13] Hengrui Gu, Kaixiong Zhou, Xiaotian Han, Ninghao Liu, Ruobing Wang, and Xin Wang. Pokemqa: Programmable knowledge editing for multi-hop question answering, 2024.

[14] Suriya Gunasekar, Yi Zhang, Jyoti Aneja, Caio César Teodoro Mendes, Allie Del Giorno, Sivakanth Gopi, Mojan Javaheripi, Piero Kauffmann, Gustavo de Rosa, Olli Saarikivi, Adil Salim, Shital Shah, Harkirat Singh Behl, Xin Wang, Sébastien Bubeck, Ronen Eldan, Adam Tauman Kalai, Yin Tat Lee, and Yuanzhi Li. Textbooks are all you need, 2023.

[15] Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. Don't stop pretraining: Adapt language models to domains and tasks, 2020.

[16] Guoxiu He, Xin Song, and Aixin Sun. Knowledge updating? no more model editing! just selective contextual reasoning, 2025.

[17] Pavan Holur, Shreyas Rajesh, David Chong, and Vwani Roychowdhury. Creating an ai observer: Generative semantic workspaces, 2024.

[18] Michael Y. Hu, Jackson Petty, Chuan Shi, William Merrill, and Tal Linzen. Between circuits and chomsky: Pre-pretraining on formal languages imparts linguistic biases, 2025.

[19] Jing Huang, Diyi Yang, and Christopher Potts. Demystifying verbatim memorization in large language models. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 10711–10732, Miami, Florida, USA, November 2024. Association for Computational Linguistics.

[20] Alexis Huet, Zied Ben Houidi, and Dario Rossi. Episodic memories generation and evaluation benchmark for large language models, 2025.

[21] Joel Jang, Seonghyeon Ye, Changho Lee, Sohee Yang, Joongbo Shin, Janghoon Han, Gyeonghun Kim, and Minjoon Seo. TemporalWiki: A lifelong benchmark for training and evaluating ever-evolving language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6237–6250, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics.

[22] Xisen Jin, Dejiao Zhang, Henghui Zhu, Wei Xiao, Shang-Wen Li, Xiaokai Wei, Andrew Arnold, and Xiang Ren. Lifelong pretraining: Continually adapting language models to emerging corpora, 2022.

[23] Tianjie Ju, Yijin Chen, Xinwei Yuan, Zhuosheng Zhang, Wei Du, Yubin Zheng, and Gongshen Liu. Investigating multi-hop factual shortcuts in knowledge editing of large language models, 2024.

[24] Feiyang Kang, Newsha Ardalani, Michael Kuchnik, Youssef Emad, Mostafa Elhoushi, Shubhabrata Sengupta, Shang-Wen Li, Ramya Raghavendra, Ruoxi Jia, and Carole-Jean Wu. Demystifying synthetic data in llm pre-training: A systematic study of scaling laws, benefits, and pitfalls, 2025.

[25] Jungo Kasai, Keisuke Sakaguchi, Yoichi Takahashi, Ronan Le Bras, Akari Asai, Xinyan Yu, Dragomir Radev, Noah A. Smith, Yejin Choi, and Kentaro Inui. Realtime qa: What's the answer right now?, 2024.

[26] John Kirchenbauer, Janny Mongkolsupawan, Yuxin Wen, Tom Goldstein, and Daphne Ippolito. Fictionalqa: A dataset for studying memorization and knowledge acquisition, 2026.

[27] Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V. Miranda, Alisa Liu, Nouha Dziri, Shane Lyu, Yuling Gu, Saumya Malik, Victoria Graf, Jena D. Hwang, Jiangjiang Yang, Ronan Le Bras, Oyvind Tafjord, Chris Wilhelm, Luca Soldaini, Noah A. Smith, Yizhong Wang, Pradeep Dasigi, and Hannaneh Hajishirzi. Tulu 3: Pushing frontiers in open language model post-training, 2025.

[28] Nathaniel Li, Alexander Pan, Anjali Gopal, Summer Yue, Daniel Berrios, Alice Gatti, Justin D. Li, Ann-Kathrin Dombrowski, Shashwat Goel, Long Phan, Gabriel Mukobi, Nathan Helm-Burger, Rassin Lababidi, Lennart Justen, Andrew B. Liu, Michael Chen, Isabelle Barrass, Oliver Zhang, Xiaoyuan Zhu, Rishub Tamirisa, Bhrugu Bharathi, Adam Khoja, Zhenqi Zhao, Ariel Herbert-Voss, Cort B. Breuer, Samuel Marks, Oam Patel, Andy Zou, Mantas Mazeika, Zifan Wang, Palash Oswal, Weiran Lin, Adam A. Hunt, Justin Tienken-Harder, Kevin Y. Shih, Kemper Talley, John Guan, Russell Kaplan, Ian Steneker, David Campbell, Brad Jokubaitis, Alex Levinson, Jean Wang, William Qian, Kallol Krishna Karmakar, Steven Basart, Stephen Fitz, Mindy Levine, Ponnurangam Kumaraguru, Uday Tupakula, Vijay Varadharajan, Ruoyu Wang, Yan Shoshitaishvili, Jimmy Ba, Kevin M. Esvelt, Alexandr Wang, and Dan Hendrycks. The wmdp benchmark: Measuring and reducing malicious use with unlearning, 2024.

[29] Zichao Li, Ines Arous, Siva Reddy, and Jackie C. K. Cheung. Evaluating dependencies in fact editing for language models: Specificity and implication awareness, 2023.

[30] Adam Liska, Tomas Kocisky, Elena Gribovskaya, Tayfun Terzi, Eren Sezener, Devang Agrawal, Cyprien De Masson D'Autume, Tim Scholtes, Manzil Zaheer, Susannah Young, et al. Streamingqa: A benchmark for adaptation to new knowledge over time in question answering models. In *International Conference on Machine Learning*, pages 13604–13622. PMLR, 2022.

[31] Jinzhe Liu, Junshu Sun, Shufan Shen, Chenxue Yang, and Shuhui Wang. Edit less, achieve more: Dynamic sparse neuron masking for lifelong knowledge editing in llms, 2025.

[32] Ruibo Liu, Jerry Wei, Fangyu Liu, Chenglei Si, Yanzhe Zhang, Jinmeng Rao, Steven Zheng, Daiyi Peng, Diyi Yang, Denny Zhou, and Andrew M. Dai. Best practices and lessons learned on synthetic data, 2024.

[33] Yu Lu Liu, Su Lin Blodgett, Jackie Chi Kit Cheung, Q. Vera Liao, Alexandra Olteanu, and Ziang Xiao. Ecbd: Evidence-centered benchmark design for nlp, 2024.

[34] Yinyi Luo, Zhexian Zhou, Hao Chen, Kai Qiu, Marios Savvides, Sharon Li, and Jindong Wang. Knowledgesmith: Uncovering knowledge updating in llms with model editing and unlearning, 2026.

[35] Kelvin Luu, Daniel Khashabi, Suchin Gururangan, Karishma Mandyam, and Noah A. Smith. Time waits for no one! analysis and challenges of temporal misalignment. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5944–5958, Seattle, United States, July 2022. Association for Computational Linguistics.

[36] Pratyush Maini, Zhili Feng, Avi Schwarzschild, Zachary C. Lipton, and J. Zico Kolter. Tofu: A task of fictitious unlearning for llms, 2024.

[37] Pratyush Maini, Skyler Seto, He Bai, David Grangier, Yizhe Zhang, and Navdeep Jaitly. Rephrasing the web: A recipe for compute and data-efficient language modeling, 2024.

[38] Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories, 2023.

[39] Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in gpt, 2023.

[40] Kevin Meng, Arnab Sen Sharma, Alex Andonian, Yonatan Belinkov, and David Bau. Mass-editing memory in a transformer, 2023.

[41] Eric Mitchell, Charles Lin, Antoine Bosselut, Chelsea Finn, and Christopher D. Manning. Fast model editing at scale, 2022.

[42] Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori Hashimoto. s1: Simple test-time scaling, 2025.

[43] Dang Nguyen, Zeman Li, Mohammadhossein Bateni, Vahab Mirrokni, Meisam Razaviyayn, and Baharan Mirzasoleiman. Synthetic text generation for training large language models via gradient matching, 2025.

[44] Team Olmo, :, Allyson Ettinger, Amanda Bertsch, Bailey Kuehl, David Graham, David Heineman, Dirk Groeneveld, Faeze Brahman, Finbarr Timbers, Hamish Ivison, Jacob Morrison, Jake Poznanski, Kyle Lo, Luca Soldaini, Matt Jordan, Mayee Chen, Michael Noukhovitch, Nathan Lambert, Pete Walsh, Pradeep Dasigi, Robert Berry, Saumya Malik, Saurabh Shah, Scott Geng, Shane Arora, Shashank Gupta, Taira Anderson, Teng Xiao, Tyler Murray, Tyler Romero, Victoria Graf, Akari Asai, Akshita Bhagia, Alexander Wettig, Alisa Liu, Aman Rangapur, Chloe Anastasiades, Costa Huang, Dustin Schwenk, Harsh Trivedi, Ian Magnusson, Jaron Lochner, Jiacheng Liu, Lester James V. Miranda, Maarten Sap, Malia Morgan, Michael Schmitz, Michal

Guerquin, Michael Wilson, Regan Huff, Ronan Le Bras, Rui Xin, Rulin Shao, Sam Skjonsberg, Shannon Zejiang Shen, Shuyue Stella Li, Tucker Wilde, Valentina Pyatkin, Will Merrill, Yapei Chang, Yuling Gu, Zhiyuan Zeng, Ashish Sabharwal, Luke Zettlemoyer, Pang Wei Koh, Ali Farhadi, Noah A. Smith, and Hannaneh Hajishirzi. Olmo 3, 2025.

[45] Yasumasa Onoe, Michael J. Q. Zhang, Shankar Padmanabhan, Greg Durrett, and Eunsol Choi. Can lms learn new entities from descriptions? challenges in propagating injected knowledge, 2023.

[46] OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, et al. Gpt-4 technical report, 2024.

[47] Core Francisco Park, Zechen Zhang, and Hidenori Tanaka. *New News*: System-2 fine-tuning for robust integration of new knowledge, 2025.

[48] Ajay Patel, Colin Raffel, and Chris Callison-Burch. Datadreamer: A tool for synthetic data generation and reproducible llm workflows, 2024.

[49] Valentina Pyatkin, Saumya Malik, Victoria Graf, Hamish Ivison, Shengyi Huang, Pradeep Dasigi, Nathan Lambert, and Hannaneh Hajishirzi. Generalizing verifiable instruction following, 2025.

[50] Siyuan Qi, Bangcheng Yang, Kailin Jiang, Xiaobo Wang, Jiaqi Li, Yifan Zhong, Yaodong Yang, and Zilong Zheng. In-context editing: Learning knowledge from self-induced distributions, 2025.

[51] Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model, 2024.

[52] Shreyas Rajesh, Pavan Holur, Chenda Duan, David Chong, and Vwani Roychowdhury. Beyond fact retrieval: Episodic memory for rag with generative semantic workspaces, 2026.

[53] Shreyas Rajesh, Pavan Holur, Mehmet Yigit Turali, Chenda Duan, and Vwani Roychowdhury. Panini: Continual learning in token space via structured memory, 2026.

[54] Shruti Rijhwani and Daniel Preotiuc-Pietro. Temporally-informed analysis of named entity recognition. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7605–7617, Online, July 2020. Association for Computational Linguistics.

[55] Cécile Rousseau, Tobia Boschi, Giandomenico Cornacchia, Dhaval Salwala, Alessandra Pascale, and Juan Bernabe Moreno. Forging time series with language: A large language model approach to synthetic data generation, 2025.

[56] Amit Rozner, Barak Battash, Lior Wolf, and Ofir Lindenbaum. Knowledge editing in language models via adapted direct preference optimization, 2024.

[57] Agam Shah, Liqin Ye, Sebastian Jaskowski, Wei Xu, and Sudheer Chava. Beyond the reported cutoff: Where large language models fall short on financial knowledge, 2025.

[58] Weijia Shi, Jaechan Lee, Yangsibo Huang, Sadhika Malladi, Jieyu Zhao, Ari Holtzman, Daogao Liu, Luke Zettlemoyer, Noah A. Smith, and Chiyuan Zhang. Muse: Machine unlearning six-way evaluation for language models, 2024.

[59] Charlie Snell, Dan Klein, and Ruiqi Zhong. Learning by distilling context, 2022.

[60] Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, Louis Rouillard, Thomas Mesnard, Geoffrey Cideron, Jean bastien Grill, Sabela Ramos, Edouard Yvinec, Michelle Casbon, Etienne Pot, Ivo Penchev, Gaël Liu, Francesco Visin, Kathleen Kenealy, Lucas Beyer, Xiaohai Zhai, Anton Tsitsulin, Robert Busa-Fekete, Alex Feng, Noveen Sachdeva, Benjamin Coleman, Yi Gao, Basil Mustafa, et al. Gemma 3 technical report, 2025.

[61] Lukas Thede, Karsten Roth, Matthias Bethge, Zeynep Akata, and Thomas Hartvigsen. WikiBigEdit: Understanding the limits of lifelong knowledge editing in LLMs. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu, editors, *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 59326–59354. PMLR, 13–19 Jul 2025.

[62] Denny Vrandečić and Markus Krötzsch. Wikidata: a free collaborative knowledgebase. *Commun. ACM*, 57(10):78–85, September 2014.

[63] Tu Vu, Mohit Iyyer, Xuezhi Wang, Noah Constant, Jerry Wei, Jason Wei, Chris Tar, Yun-Hsuan Sung, Denny Zhou, Quoc Le, and Thang Luong. Freshllms: Refreshing large language models with search engine augmentation, 2023.

[64] Peng Wang, Zexi Li, Ningyu Zhang, Ziwen Xu, Yunzhi Yao, Yong Jiang, Pengjun Xie, Fei Huang, and Huajun Chen. Wise: Rethinking the knowledge memory for lifelong model editing of large language models, 2024.

[65] Yiwei Wang, Muhao Chen, Nanyun Peng, and Kai-Wei Chang. Deepedit: Knowledge editing as decoding with constraints, 2024.

[66] Yizhong Wang, Hamish Ivison, Pradeep Dasigi, Jack Hessel, Tushar Khot, Khyathi Raghavi Chandu, David Wadden, Kelsey MacMillan, Noah A. Smith, Iz Beltagy, and Hannaneh Hajishirzi. How far can camels go? exploring the state of instruction tuning on open resources, 2023.

[67] Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark, 2024.

[68] Zhilin Wang, Yi Dong, Olivier Delalleau, Jiaqi Zeng, Gerald Shen, Daniel Egert, Jimmy J. Zhang, Makesh Narsimhan Sreedhar, and Oleksii Kuchaiev. Helpsteer2: Open-source dataset for training top-performing reward models, 2024.

[69] Haoze Wu, Yunzhi Yao, Wenhao Yu, and Ningyu Zhang. Recode: Updating code api knowledge with reinforcement learning, 2025.

[70] Hao Xiong, Chuanyuan Tan, and Wenliang Chen. Is fine-tuning an effective solution? reassessing knowledge editing for unstructured data, 2025.

[71] Zitong Yang, Neil Band, Shuangping Li, Emmanuel Candès, and Tatsunori Hashimoto. Synthetic continued pretraining, 2024.

[72] Junsang Yoon, Akshat Gupta, and Gopala Anumanchipalli. Is bigger edit batch size always better? – an empirical study on model editing with llama-3, 2024.

[73] Ningyu Zhang, Yunzhi Yao, Bozhong Tian, Peng Wang, Shumin Deng, Mengru Wang, Zekun Xi, Shengyu Mao, Jintian Zhang, Yuansheng Ni, Siyuan Cheng, Ziwen Xu, Xin Xu, Jia-Chen Gu, Yong Jiang, Pengjun Xie, Fei Huang, Lei Liang, Zhiqiang Zhang, Xiaowei Zhu, Jun Zhou, and Huajun Chen. A comprehensive study of knowledge editing for large language models, 2024.

[74] Ruiqi Zhang, Licong Lin, Yu Bai, and Song Mei. Negative preference optimization: From catastrophic collapse to effective unlearning, 2024.

[75] Ce Zheng, Lei Li, Qingxiu Dong, Yuxuan Fan, Zhiyong Wu, Jingjing Xu, and Baobao Chang. Can we edit factual knowledge by in-context learning?, 2023.

[76] Jonathan Zheng, Alan Ritter, and Wei Xu. Neo-bench: Evaluating robustness of large language models with neologisms, 2024.

[77] Shaochen Zhong, Yifan Lu, Lize Shao, Bhargav Bhushanam, Xiaocong Du, Yixin Wan, Yucheng Shi, Daochen Zha, Yiwei Wang, Ninghao Liu, Kaixiong Zhou, Shuai Xu, Kai-Wei Chang, Louis Feng, Vipin Chaudhary, and Xia Hu. MQuAKE-remastered: Multi-hop knowledge editing can only be advanced with reliable evaluations. In *The Thirteenth International Conference on Learning Representations*, 2025.

[78] Zexuan Zhong, Zhengxuan Wu, Christopher D. Manning, Christopher Potts, and Danqi Chen. Mquake: Assessing knowledge editing in language models via multi-hop questions, 2024.

[79] Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. Instruction-following evaluation for large language models, 2023.

A Impact Statement

This paper explores novel methodologies for knowledge updating in large language models (LLMs), with the goal of addressing temporal knowledge drift and improving model safety and reliability. Such improvements are particularly important for users who rely on LLMs in real-time, high-stakes scenarios, such as search engines, live event tracking, or emergency response systems. By enabling models to incorporate new facts accurately, we aim to reduce the risk of outdated or incorrect information being presented to users.

Our dataset consists entirely of fictional events designed to be realistic, including topics with serious implications, such as natural disasters (with casualty counts) and economic crises in various regions. To ensure plausibility, we leverage real events as prompts to guide the model in generating fictional events with credible outcomes, locations, and details. While this process may still introduce potential biases, each event is carefully reviewed by in-house annotators and one of the authors to minimize harmful content—for example, by removing culturally insensitive or biased names of individuals.

Generating realistic synthetic knowledge also can carry potential misuse risks, such as the creation of convincing misinformation or fabricated "facts" that could be used to manipulate downstream models. Our event topics (e.g., natural disasters, sporting events, movies) are large-scale and public in nature rather than targeting specific real entities, which limits the risk of misuse for targeted misinformation or harassment. To further mitigate misuse, we explicitly flag all generated events as synthetic/fictional in the released dataset via metadata tags and clear naming conventions, restrict release to research-oriented licenses, and recommend that any derivative use continue to clearly label content as synthetic to prevent conflation with real-world facts.

A key finding of our work is that, although our methodology reduces temporal knowledge drift, it slightly decreases the model's general LLM capabilities. Nevertheless, model performance remains comparable, and we observe that the largest degradations occur in out-of-domain topics like math and code. These observations highlight the trade-offs between knowledge updating, reasoning capability, and model reliability. Future research could look at knowledge updating while maintaining technical knowledge, like code capabilities.

We emphasize that the deployment of updated LLMs should be done responsibly. Potential risks include the propagation of errors in real-time decision-making or the inadvertent reinforcement of subtle biases in synthetic data. Future research should explore the societal impact of such models, including user studies in urgent scenarios, like live sporting events or impending natural disasters. Moreover, ongoing work should investigate techniques to further ensure fairness, transparency, and accountability when updating model knowledge, particularly when models interact with diverse populations or sensitive information.

Overall, our work aims to enhance the fidelity and utility of LLMs in dynamic real-world contexts, enabling models to provide more accurate, timely, and actionable information. Beyond improving reliability, knowledge-updated LLMs have the potential to empower decision-making in critical domains, support evidence-based policy, and increase public trust in AI systems when deployed

responsibly. By carefully integrating new information, these models can contribute to more equitable access to knowledge, help mitigate human error in high-stakes scenarios, and assist underrepresented communities with tailored insights. Furthermore, applications in healthcare, disaster response, education, and governance could directly improve human well-being and societal outcomes. These opportunities highlight the importance of continued research into transparent updating procedures, robust auditing mechanisms, and participatory evaluation methods that engage diverse stakeholders, ensuring that knowledge-updated LLMs provide socially beneficial, ethical, and trustworthy support across a wide range of domains.

B Limitations

Dataset scope and design. While PARALLELEVENTS covers only 6 primary event topics, each event is highly complex; for instance, a hypothetical FIFA World Cup event contains over 40,000 true editable facts, including non-consequential details such as players' spouses. Limited automation is a deliberate design choice to ensure events remain fictional, realistic, and internally consistent, with construction requiring approximately 350 hours overall. To address potential bias, we employ manual verification and diversity augmentation, evenly sampling from automatically produced parameters such as geographic region and severity to prevent overrepresentation of particular entities. Future work should look into extensions of this parallel universe for more events, such as company mergers or the creation of a new commercial product.

PARALLELEVENTS spans the years 2030–2035 but is designed with refreshability in mind: for each event, we can systematically advance the associated years or dates through simple functional graph parsing, allowing us to push the dataset's temporal end date forward as needed. For example, a fictional event like "Hurricane Alicia occurred on August 4th, 2030" can be updated to "September 3rd, 2043," allowing flexible experimentation across temporal contexts and supporting temporal insertions for new LLMs. Additionally, during dataset construction we compiled a structured list of valid parameters for each event type (e.g., locations, winners, attendance figures, economic costs), which allows us to shuffle and generate new permutations of events that still satisfy real-world constraints, producing fresh event instances not present in any prior release. Together, these mechanisms allow PARALLELEVENTS to be periodically refreshed, mitigating contamination risk as newer pretraining corpora are released.

Training methodology. We only explore using DPO in the offline setting, due to its unexplored application for factual insertions of temporal events; this offline setup also avoids the challenge of training a reward model for hypothetical future events, though dynamic updates can be handled by periodically refreshing the training set. Future work should investigate online settings, such as using GRPO for updating codebases [69], to enable more dynamic, real-time knowledge integration.

SYNAPSE requires the generation of high-quality synthetic data for updating a model's knowledge. While generating high-quality synthetic SFT and preference data incurs some overhead, this cost is substantially lower than continual pretraining, which requires trillions of training tokens. Table 4 provides the statistics for the amount of data needed to be generated for each training split, the generation time, and the training time. For moderate factual insertions, this can be done in a day. We also examine the effect of varying the teacher model used to train SYNAPSE (Table 10 and Table 11), and find that teacher strength does affect SYNAPSE's performance. However, even with weaker teacher models, SYNAPSE outperforms the strongest baselines in our benchmark.

Evaluation scope. We provide analyses of knowledge editing on this dataset, evaluating settings with 25, 150, 542, and 1,536 inserted facts, as well as a 100-fact setting on MQUAKE. These choices are partly motivated by the computational and data-generation costs of larger-scale interventions, which require substantially longer training times. Importantly, we find that, on average, only 20–50 *consequential* facts per event require insertion (e.g., those that affect key outcomes or information like the winner of the World Cup), suggesting that our methodology is well suited for real-time event editing. In this regime, our approach consistently outperforms in-context learning baselines. Due to the computational cost of training models with different subsets of training data, including abstention data and general preference mixtures, hyperparameters, and ablations, we only explore ablations on the 150-fact setting. Additionally, we report results only for LLaMA-3.1-8B, given its superior performance and editability compared to Olmo-3.

Sequential updates and catastrophic forgetting. Our current evaluation of SYNAPSE focuses on a batch insertion setting, where the model is updated once with a set of 10 to 1,536 synthetic facts and then evaluated on general benchmarks (e.g., MMLU-Pro, PopQA) alongside the inserted knowledge. This design is consistent with prior knowledge-insertion work, but it leaves open the question of how SYNAPSE behaves under multi-round sequential updates, where new facts are inserted incrementally over time. In particular, we have not measured whether earlier-inserted facts degrade as additional rounds of updates are applied. Evaluating SYNAPSE under sequential, multi-round updates, including tracking retention of both earlier synthetic insertions and general benchmark performance across rounds, is an important direction for future work.

Scale and real-world applicability. Our evaluation of SYNAPSE is conducted in fully synthetic settings, which may not fully capture the challenges of larger-scale or real-world continual learning. Factual performance decreases as the number of new facts grows (10 to 1,536 facts), consistent with other knowledge-insertion work, though our instruction-tuning preference set helps retain instruction-following ability. We expect this degradation to compound further when knowledge insertions are combined with task-specific fine-tuning or scaled to hundreds of events. Real-world continual learning also introduces challenges beyond our synthetic setting, such as noisy or contradictory updates, which likely require a contradiction-resolution step before integration. We view scaling SYNAPSE to noisier knowledge domains as important future work.

C Extended Related Work

Knowledge Updating. Prior work has identified model degradation due to temporal drift across multiple dimensions, including neologisms [76], named entities [54, 33, 61], recent facts [63, 25, 78], and counterfactual knowledge [39, 29, 73, 78, 77]. Closely related work also constructs hypothetical-but-plausible future events, but evaluates whether a model internalizes a given fact and its immediate implication in isolation [47]; in contrast, our events are embedded within a dense Wikidata-style knowledge graph, enabling us to evaluate whether a fact's consequences are correctly propagated across a broader web of existing knowledge.

A substantial body of work has explored knowledge editing approaches, including localized model edits via auxiliary hypernetworks [41] or direct modification of transformer weights associated with specific facts [39, 40, 31, 64, 9]. In-context learning methods for injecting updated knowledge at inference time [75, 50] and retrieval- or graph-based mechanisms that provide relevant facts from structured knowledge sources [78, 13, 65, 77] have also been explored. More recently, reinforcement learning has been investigated for integrating knowledge updates across code APIs [69], general factual knowledge, and counterfactuals [56], alongside complementary strategies such as single-layer fine-tuning [70] and masked text fine-tuning [73]. There has also been prior work in studying knowledge unlearning [74, 36, 8, 58, 28] acquisition and memorization in synthetic texts [26, 19, 3]. In research closely related to ours, data generation has been utilized to finetune models for continual learning on general benchmarks [34] and standalone news articles [47]. Despite this progress, existing work largely focuses on correcting simple or isolated knowledge updates, like singular news articles, with no systematic study of temporal drift induced by complex, realistic future events.

A separate line of work models temporally-structured experience explicitly as episodic memory, rather than relying on undifferentiated context windows or parametric weight updates, with benchmarks and non-parametric structured-memory frameworks proposed for multi-hop reasoning, link prediction, and event anticipation over such memories [20, 52, 17, 53]. Context distillation methods, which internalize in-context information into model parameters via self-generated fine-tuning data, have also been studied for knowledge insertion, including protocols for closing the gap between fine-tuning and in-context learning [59, 47].

Prior work has also explored other facets of knowledge updating, including knowledge forgetting via earlier temporal cutoffs that test a model's ability to provide time-conditioned information [10], as well as real-time question answering using lifelong snapshots of Wikipedia [63, 25, 61]. While these interfaces provide useful datasets for studying LLM learning dynamics through in-context learning or evaluating isolated, one-off updates to factual triples, they fail to capture the complex, interdependent, and evolving nature of event-based knowledge updates that will occur in the future.

IKE	Total Accuracy	Single Hop	Multi-Hop	Causal
LLaMA-3.1-8B-Instruct				
10	57.99%	91.23%	31.62%	38.56%
32	64.23%	95.52%	43.96%	41.39%
64	63.62%	93.84%	38.05%	38.05%
128	62.57%	93.09%	40.36%	42.67%
542	62.41%	92.91%	53.47%	29.31%
Olmo-3-7B-Instruct				
10	58.22%	92.72%	31.62%	37.28%
32	59.59%	94.03%	28.02%	43.70%
64	58.98%	93.10%	26.99%	43.96%
128	59.36%	93.10%	29.82%	42.41%
542	55.86%	87.87%	41.39%	26.22%

Table 3: PARALLELEVENTS accuracy for IKE when prompted with differing amounts of retrieved facts. For both models, providing 32 facts yields the highest overall accuracy.

# of facts	# of Events	# of Tokens	Generation time (hours)	Training time (hours, 2 H100s)
20	1	89,099	0.50	0.4
150	15	1,336,471	6.50	2.2
542	29	3,738,524	18.17	8.1
1,536	41	10,438,174	50.74	22.5

Table 4: Generation and Training time statistics for SYNAPSE across 20, 150, 542, and 1,536 fact insertion cases. Both the preference dataset and the long form text are included for each training split. Tokens are measured using the LLaMA-3.1 Tokenizer.

Synthetic Text Generation. There has been substantial work exploring the synthetic generation of pre-training data for small language models [7], domain adaptation in general tasks [15, 71], and code [14]. These methods often iteratively generate new data from a small seed set of real examples. Other approaches explore synthetic data generation via rephrasing human-written text [37], training models on linguistic rules derived from formal languages [18], and prompting models to encourage more diverse generations [48, 4]. Prior research has also studied the generation of synthetic post-training data for instruction following [27, 66], alignment [11], and reasoning [42]. Analyses of synthetic data generation largely focus on scaling laws [24], data quality [32], and data novelty [71]. Some work utilizes synthetic data throughout the entire training pipeline, spanning both pre-training and post-training, particularly for code-based models [1]. Most closely related to our work, prior research has leveraged synthetic data generation for time-series data [55] and novel event detection [6].

Closer still, synthetic generation has recently been used to construct benchmarks and training signal specifically for knowledge updating: [47] use context distillation over synthetically generated documents to integrate new knowledge into model weights; [34] synthesize edits to probe model editing and unlearning; and [20] construct synthetic episodic memories to evaluate LLMs' temporal and contextual recall. These approaches, however, target synthetically invented or static facts rather than complex, grounded, real-world events. To the best of our knowledge, no prior work has explored the synthetic generation of temporal event data grounded in real-world occurrences, particularly for updating LLMs with knowledge of such events.

D Benchmark Details

Figure 2 reports Wikipedia page counts for subject–object co-occurrence pairs (s, o) from factual triples in existing benchmarks and PARALLELEVENTS, serving as a proxy for how strongly such rigid facts are represented in pre-training data. We find that entity pairs in MQUAKE-CF and COUNTERFACT [39] frequently co-occur across large numbers of documents, with 38.43% and 10.16% of pairs appearing in more than 1,000 Wikipedia pages, respectively. For example, the edit $(\text{United Kingdom}, \text{part of}, \text{Europe}) \rightarrow (\text{United Kingdom}, \text{part of}, \text{Oceania})$ involves entities that co-occur in 120,340 Wikipedia articles, indicating a strong pre-existing association that makes downstream queries inconsistent or difficult to resolve after editing; similarly, in COUNTERFACT, the edit $(\text{Mozambique}, \text{continent}, \text{Africa}) \rightarrow (\text{Mozambique}, \text{continent}, \text{Antarctica})$ corresponds to 34,815 co-occurring articles. Empirically, across 42 related questions per edited fact, LLaMA-3.1-8B-Instruct achieves only 17.86% accuracy after editing with in-context prompting, suggesting strong

conflicts between edits and pre-training knowledge. In contrast, PARALLELEVENTS contains far fewer highly co-occurring entity pairs. For instance, a synthetic event in which South Whittier becomes an incorporated city in Los Angeles County has the factual triple (South Whittier, incorporated city, Los Angeles County), and South Whittier and Los Angeles County only co-occur in 123 shared articles. Thus, the future worlds of ParallelEvents reduce pre-existing associations and thereby mitigate unintended inconsistencies introduced by factual edits.

In total, our dataset covers natural disasters, general elections, sporting events, economic crises, and new city incorporations. The full dataset includes one creative work with a movie; one global economic crisis; four city incorporations in the United States and China; six one-off natural disasters, including a blizzard, a city fire, a hurricane, a sinkhole, and two tornadoes; and ten cause-and-effect disaster pairs. These include six random events—namely an earthquake and conflagration, an earthquake and derailment, an earthquake and oil spill, a thunderstorm and flash floods, a volcano and landslide, and a volcano and tsunami—and four seasonal events, including a blizzard and avalanche, a drought and wildfire, flooding and landslide, and a typhoon and flooding. The dataset also includes three sporting events: the 2030 NBA season, the 2035 NFL season (including the Super Bowl and halftime show performance), and the full 2034 FIFA World Cup, including all players, coaches, and referees. Players in the FIFA World Cup are split evenly between real current players and fictional names. We also cover one UK general election, with both a close result and landslide result variant, where the leaders of all participating political parties are fictional individuals. The dataset only contains entities that participate in at least one factual triple requiring an update.

For consistency checks, 2 in-house annotators (including an author) verified events for realism, entity-level attribute issues, uniqueness, temporal consistency, and stereotypes/bias in names and countries, enforcing unique names, consistent tournament brackets, and valid age statistics for real players. Election vote counts used human-guided generation to maintain realistic ranges for existing parties, as fully synthetic generation of these numerical attributes was inconsistent. We doubly annotate 20% of the events and calculate the inter-annotator agreement of selecting a JSON field for editing to be Cohen's $\kappa = 0.8298$. 26.67% of events were discarded and regenerated (incoherence, unrealism, excessive similarity), and a further 14.3% of events generated during knowledge graph construction were filtered for being too similar to prior events. 35% of entities in accepted events were manually edited for logical consistency, most commonly for entity uniqueness (e.g., duplicate coaches), temporal ordering, and invalid tournament brackets. In the final dataset, the most frequent relation is "start time" (29,324), and the most frequent entity is "association football" (2,627).

Potential stereotypes or negative biases associated with names and countries are also reviewed by an in-house annotator. For the election scenarios, we utilize human labor combined with random generation to determine vote counts for each party while maintaining realistic ranges for existing parties, as fully synthetic generation of these numerical attributes was found to be inconsistent. For relations, the most frequent relation is "start time" (29,324), and the most frequent entity is "association football" (2,627). For generated knowledge graphs, 14.3% of events are filtered for being too similar to prior events. 35% of entities in the dataset are manually edited for logical consistency.

For evaluation, each fact-insertion setting uses a different subset of events. The 20-fact setting explores a single natural disaster and its effects; the 150-fact setting explores 15 natural disasters; the 542-fact setting explores 29 events, consisting of 26 natural disasters and three sporting events plus the halftime show; and the 1,536-fact setting explores all 41 events. Multi-hop questions in our benchmark average 3.06 hops (2/3/4-hop: 26.9% / 39.9% / 26.0%), while causal questions average 1.66 hops (a step in a chain of actions) across 40+ consequence types. We utilize LLM-as-a-judge to evaluate all knowledge-insertion methods: two LLM judges (LLaMA-3.1-70B-Instruct, GPT-OSS-20B) independently assess semantic equivalence, with disagreements (5.67% of answers) sent to human adjudication. On a separate 500-pair sample, human agreement with the judge consensus reached $\kappa = 0.937$, indicating strong agreement.

E Baseline Implementation Details

The finetuning baselines are trained using the same hyperparameters as the SFT phase of SYNAPSE. Namely, we utilize 8 A40 GPUs, train for 10 epochs, use a warmup ratio of 0.06, a weight decay of 0.1, and a learning rate of 1×10^{-5}. All models are finished training within 8 hours.

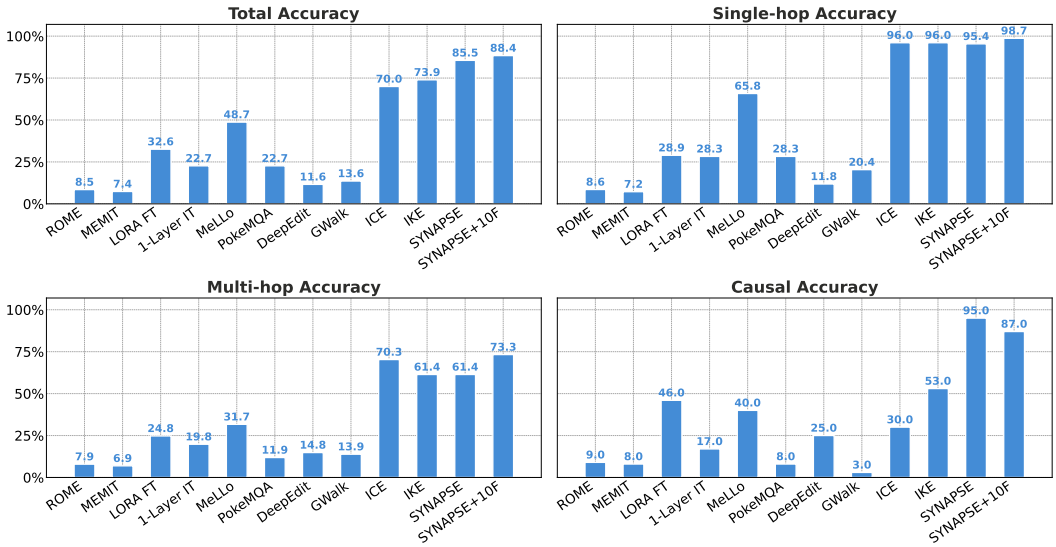

Figure 7: Question answering performance after inserting 150 facts into LLaMA-3.1-Instruct. Results are reported for **Total** accuracy, **Single**-hop, **Multi**-hop, and **Causal** accuracy (reasoning over event consequences), rounded to the nearest percent. SYNAPSE **with 10 retrieved facts achieves the highest accuracies across all question types.**

Model Name	Overall Avg.	Single-Hop	Multi-Hop	Causal
LLaMA-3.1-8B-Instruct				
ROME [39]	7.63%	6.72%	4.88%	13.37%
MEMIT [40]	9.22%	11.01%	5.14%	12.85%
WISE [64]	6.87%	7.46%	2.31%	12.60%
AlphaEdit [9]	1.53%	0.93%	1.03%	1.03%
SYNAPSE	74.11%[2]	87.50%	*44.22%*[3]	**84.32%**[1]
SYNAPSE **+ 10 Facts**	**79.83%**[1]	**98.51%**[1]	**54.76%**[1]	76.86%[2]

Table 5: Results of Model Editing baselines on the 542 fact-insertion case for LLaMA-3.1-8B Instruct. Model editing methods as a whole perform poorly on PARALLELEVENTS.

For the in-context baselines, we utilize all hyperparameters in their respective setups, including using their respective retrieval models like `Facebook Contriever`. Table 3 reports the results for IKE using {10, 32, 64, 128, 542} retrieved facts for LLaMA-3.1-8B-Instruct and OLMo-3-7B-Instruct. We focus on the setting with 32 retrieved facts, as it achieves the highest overall accuracy across all configurations. For fast inference, all methods are generated with VLLM.

We also explore model editing with ROME [41], MEMIT [41], WISE [64], and AlphaEdit [9] in the 542-fact case and report results in Table 5. We adopt hyperparameters from prior work for LLaMA-3.1-8B [72]. We do not report model editing results for Olmo-3-8B-Instruct or Gemma-3-4B-Instruct due to the model's recent release, incompatibilities with the existing software stack, and the potential time required to identify optimal hyperparameters, such as the model editing layer. We utilize the EasyEdit repository [73] for all of these methods.

F SYNAPSE Implementation Details

For the generation of long-form text and preference datasets, SYNAPSE conditions only on the set of natural-language facts, similar to all other baselines. No method has access to the underlying graph in PARALLELEVENTS. We generate multiple news article styles and content about the event, including incremental updates, retrospective analyses, and articles with both optimistic and pessimistic tones. When generating the preference data, single-hop questions are constructed by explicitly rephrasing the facts. Causal questions in the DPO set are also generated through multiple prompts to elicit logical consequences that would emerge from an event. For multi-hop questions, we utilize sampled random walks. Utilizing the graph structure of the event, we identify relations r_{hop} that are explicitly relations between entities and perform random walks on the graph to obtain fact chains. These fact chains are provided to the model to construct the multi-hop question and generate the chain-of-thought rationale in the winning response. The losing response is explicitly created by an incorrect jump

in the fact chain. We explicitly define new final relations compared to the evaluation set, ensuring that the training set does not have contamination with the test questions. We also empirically verify that the questions are disjoint between the training and evaluation sets. We find that the synthetically generated questions in the DPO set overlap with evaluation questions in only 1.79% of instances.

During synthetic dataset creation, verification filtering is conducted with the teacher model. We leverage both the knowledge graph and an evolving set of facts maintained in a knowledge base. For the long-form text, we first prompt the model to extract relevant facts so that we can append them to the knowledge base. Then, we provide the knowledge base to the model and instruct it to remove inconsistent facts and rephrase the sentences so that they remain consistent with the growing fact base. Overall, 30.89% of paragraphs in the long-form text are rewritten to maintain alignment with the knowledge graph and knowledge base. For preference data, GPT-4.1 is prompted to provide a binary decision for whether a (question, preferred) pair is logical, given the set of event facts relevant to the question-answer pair and the fact base extracted from the long text. Any question answer pairs that are unlikely, incorrect, inconsistent (e.g., if insurance premiums increase in September 2032, no other instance may claim they remain stable), or unanswerable (e.g., if there is no reference answer to a question asking about the mayor in 2035) are filtered out. Overall, 1.01% of causal preference pairs and 32.33% of multi-hop preference pairs are filtered out.

To mitigate semantic distortion and framing bias in the synthetic data, we employ a combination of diversity augmentation and post-generation validation. During generation, LLMs first produce a structured set of event parameters—such as geographic region, severity, and contextual conditions—which are evenly sampled and enforced to prevent overrepresentation of particular entities or perspectives. This includes producing balanced narratives across event outcomes (e.g., coverage of both winning and losing teams in sports events, or neutral reporting before and after disasters) and across multiple reasoning structures and question types, preventing consistent framing biases. The verification step also reduces bias by removing unverifiable or hallucinated details. To further reduce the tendency to overgeneralize or respond confidently in uncertain cases, 40% of generated samples involve unrelated events or unanswerable queries, encouraging the model to abstain when information is insufficient.

For SYNAPSE, we set attention dropout to 0.1 for both models during next-token prediction. We utilize a learning rate of 1×10^{-5}, and models are trained for 10 epochs. Additionally, we train the LM head in our experiments. We use a warmup ratio of 0.06, a weight decay of 0.1, and train using DeepSpeed. We then train the model using preference learning with a sigmoid loss, set $\beta = 0.1$, and add an auxiliary cross-entropy loss on the preferred responses with a weight of 0.1. The model is trained for 5 epochs with a learning rate of 1×10^{-6}. For Olmo-3-7B-Instruct, we increase the number of epochs to 10. Based on empirical analyses, Olmo-3-7B is more extensively instruct-tuned compared to LLaMA-3.1-8B. Consequently, behaviors such as chain-of-thought rationales in the winning response emerge more quickly in LLaMA-3.1-8B, whereas some training setups for Olmo-3-Instruct do not exhibit these behaviors even after 5 epochs. We utilize LLaMAFactory for both models, setting Olmo-3's configuration to Olmo-2 to ensure software compatibility. For preference learning, SYNAPSE is first trained on our event-specific preference dataset, and then trained on the general preference dataset for realignment.

Both setups in SYNAPSE are evaluated with the development set of our parallel universe question set, where we select the best-performing epoch. For the general preference set, we utilize a small sample of MMLU-Pro data as the development set. We train with either 8 A40 GPUs or 2 H100 GPUs in all fact settings. Table 4 provides the number of tokens, generation time, and training time needed for all training splits. SYNAPSE's framework relies on pre-generated synthetic data grounded in verified facts, which prioritizes safety and controlled learning. While this restricts immediate updates, data generation is relatively efficient: inserting hundreds of facts and training can be completed within 26 hours for moderate scales as seen in the table. This offline setup enables careful study of how much data is needed for knowledge integration and abstention behavior, following prior work [27, 44].

G Evaluation Details

For our question-answering dataset, we utilize LLaMA-3.1-70B as the LLM-as-a-judge. All evaluations are run twice, with any disagreements being resolved through human adjudication. To determine the effectiveness of LLM-as-ajudge, we compare annotator agreement with an author and the LLM. On 500 random QA pairs, human agreement with LLM-as-a-judge achieves Cohen's κ = 0.937,

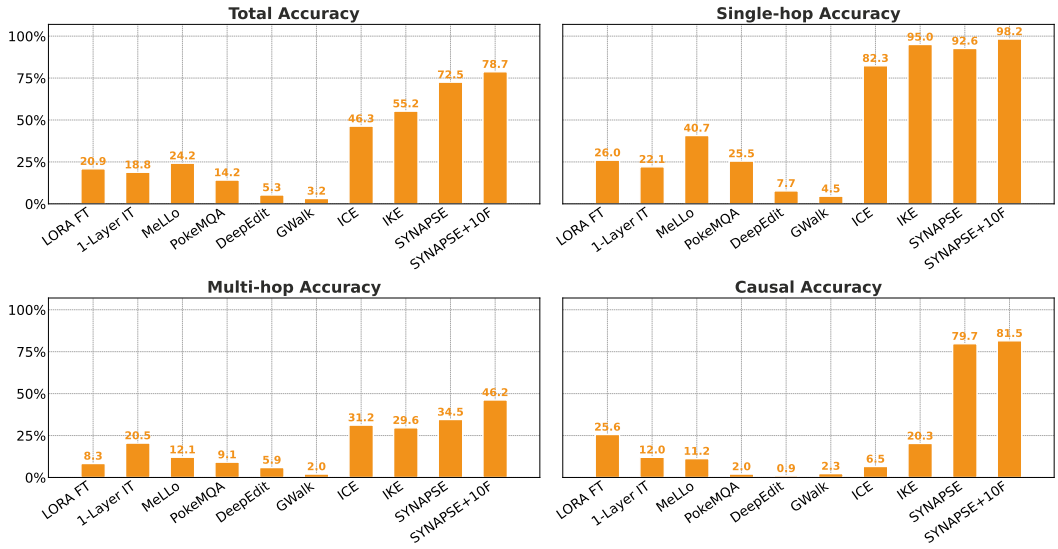

Figure 8: Question answering performance after inserting 1,536 facts into LLaMA-3.1-Instruct. Results are reported for **Total** accuracy, **Single**-hop, **Multi**-hop, and **Causal** accuracy (reasoning over event consequences), rounded to the nearest percent. SYNAPSE **with 10 retrieved facts achieves the highest accuracies across all question types.**

Topic	Number	ICE	IKE	Synapse	Synapse+10 Facts
Disaster	25.54%	46.59%	62.91%	65.13%	81.90%
Sport	35.51%	46.21%	57.10%	74.92%	80.36%
City	17.09%	38.36%	40.13%	76.05%	70.29%
Election	11.75%	53.55%	57.74%	73.87%	77.42%
Movie	5.76%	58.55%	56.58%	65.79%	81.58%
Crisis	4.36%	40.87%	46.09%	69.57%	80.00%

Table 6: Topic breakdown of question answering accuracy ICE, IKE, SYNAPSE, and SYNAPSE with 10 facts for inserting 1,536 facts into the model. SYNAPSE shows the least amount of question answering variability across question topic.

indicating strong agreement with LLM judgments. For the abstention set, an author annotates the abstention rate. For MMLU-Pro, IFEval, and IFBench, we follow the standard evaluation methodology as presented in their paper [67]. For PopQA, we utilize string matching with the provided answer set and its aliases, reporting question-answering accuracy. Evaluations are conducted on the full test sets of all general datasets.

H Additional Results

Table 5 compares model-editing methods with SYNAPSE on LLaMA-3.1-8B-Instruct for the 542 fact insertion setting; we do not report results for other models due to unavailable hyperparameters. We evaluate ROME [41], MEMIT [41], WISE [64], and AlphaEdit [9]. Overall, model editing performs poorly, achieving only 6.31% total accuracy, although it attains relatively higher performance on causal questions. Inspecting outputs, we find these methods fail to incorporate new events and often produce incoherent responses, likely due to the prevalence of unseen entities in our benchmark. Because model editing modifies existing entity associations, it is largely ineffective in this setting. Model editing methods also increase hallucinations, with a low abstention rate of 4.85% on unanswerable questions in our dataset.

Figure 7 presents the full results for inserting 150 facts into LLaMA-3.1-8B. Due to the substantial computational cost of scaling experiments across all models, we report scaling results only for LLaMA-3.1-8B. Overall trends remain consistent with other settings, with SYNAPSE outperforming all baselines, largely driven by significant gains in causal reasoning accuracy. SYNAPSE achieves an overall accuracy of 85.55% and a causal accuracy of 95.00% for 150 facts.

Performance across all methods is markedly higher than in the 542-fact insertion setting, as expected. Among the baselines, ICE and IKE perform best, achieving an average total accuracy of 71.96%. Finetuning baselines exhibit performance comparable to graph-traversal methods, generally charac-

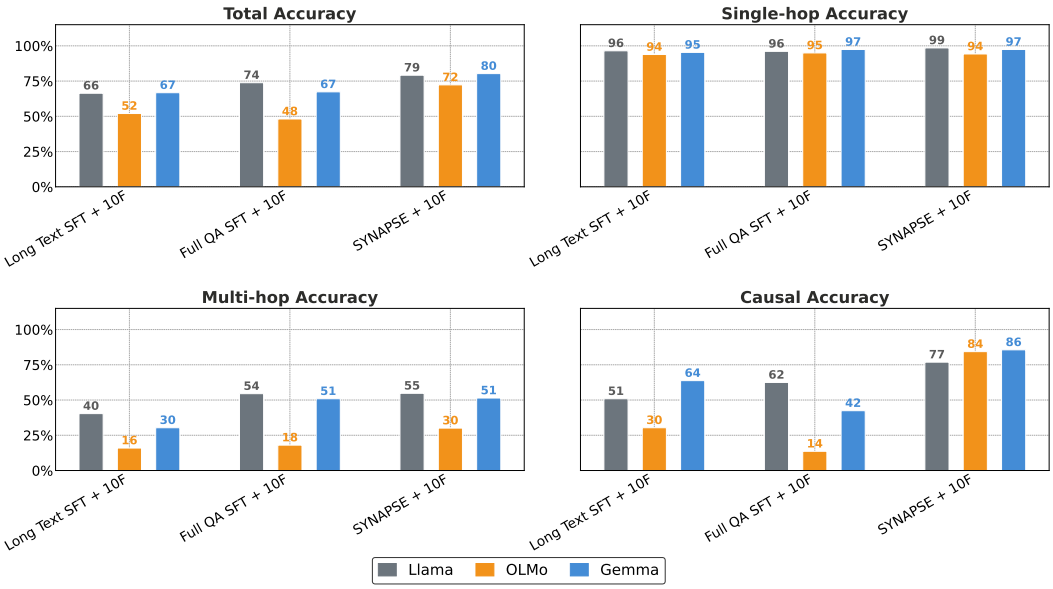

Figure 9: Accuracy of hybrid (Finetuning + RAG) methods across Llama, OLMo, and Gemma, broken down by question type. SYNAPSE + 10 Facts matches or exceeds the fine-tuned baselines under identical retrieval augmentation, with the largest margins on Multi-hop and Causal questions.

terized by high abstention rates and relatively low total accuracy. In contrast, ROME and MEMIT are the worst-performing methods, with an average total accuracy of 7.94% and an average abstention rate of 17.15%.

For 150 facts, we also test a factual unlearning method, SimNPO [8], which utilizes negative preference optimization to unlearn information while having a retain set for learned information. We utilize the dispreferred set and preferred set of SYNAPSE data as unlearned and retain sets, respectively. We find that model performance on PARALLELEVENTS is comparable to SYNAPSE, at 82.44% overall, 92.76% for single-hop, 56.44% for multihop, and 93.00% for causal questions. However, MMLU-Pro general accuracy decreases to 14.86%, and the abstention rate on PARALLELABSTAIN is only 52.86%, indicating unwanted side effects with this unlearning method.

Figure 8 presents the full results for inserting 1,536 facts into LLaMA-3.1-8B. SYNAPSE achieves an overall accuracy of 72.50% accuracy and a causal accuracy of 79.7% for 1,536 facts. Providing 10 facts in the prompt to SYNAPSE further improves performance across setups, increasing model performance to 78.70% overall, 98.2% for single-hop questions, 46.2% for multi-hop questions, and 83.50% for causal questions.

To compare with the hybrid setup of SYNAPSE plus 10 retrieved facts, we additionally ran matching hybrid setups for our fine-tuned baselines (Long Text SFT and Full QA SFT), pairing each with the same retrieved facts used for SYNAPSE. As shown in Figure 9, SYNAPSE outperforms the baseline hybrids across all three base models (Llama, OLMo, Gemma), with the largest gains concentrated in the more difficult multi-hop and causal reasoning categories. This indicates that SYNAPSE's advantage is not simply subsumed by retrieval augmentation and persists when all methods are given equal access to retrieved context.

H.1 Comparing In-Context Learning with SYNAPSE

Table 6 presents question answering accuracy across ICE, IKE, SYNAPSE, and SYNAPSE with 10 retrieved facts for the 1,536 fact editing case on LLaMA-3.1-8B-Instruct. Sport questions constitute the largest share at 35.51% of all questions, reflecting the breadth and complexity of the 2034 FIFA World Cup event. ICE and IKE exhibit substantial variability across topics, with accuracy fluctuations of 20.19% and 22.78%, respectively, whereas SYNAPSE and SYNAPSE+10 remain considerably more consistent, with fluctuations of only 10.92% and 11.61%. Across all topic categories, both SYNAPSE models outperform their in-context learning counterparts.

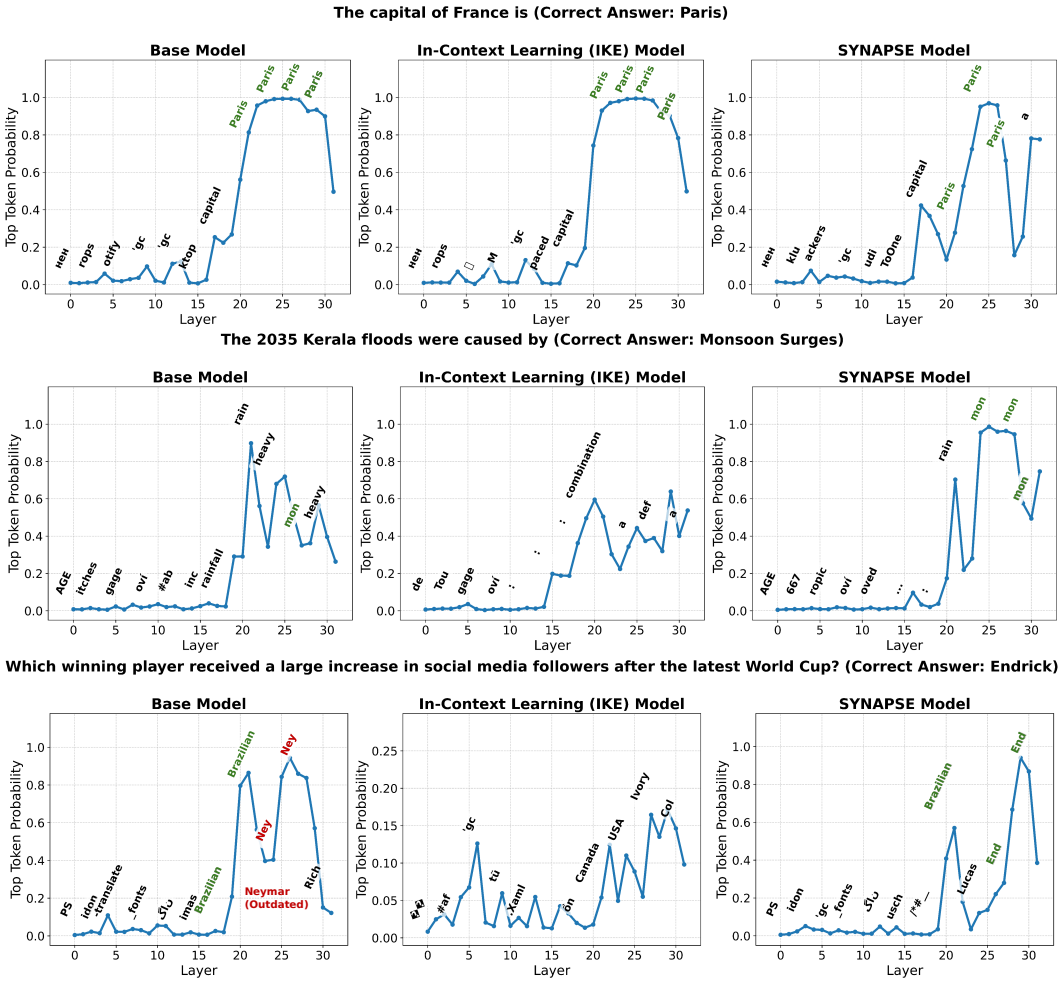

Figure 10: Token probabilities for a static question, single hop question, and causal question in the PARALLELEVENTS benchmark, comparing the base model, an in-context learning model, and SYNAPSE. Only SYNAPSE consistently produces reasonable answers about event consequences after fact insertions.

We perform an error analysis comparing IKE and ICE with SYNAPSE. The 14.23% gain in model performance largely comes from gains on causal questions and some multi-hop questions, as SYNAPSE better encodes these question dynamics. While SYNAPSE shows some rates of abstaining on relevant questions, ICE and IKE overwhelmingly abstain on causal questions. Among the questions where in-context retrieval performs better, 64.5% are single-hop queries involving exact numeric recall (e.g., event attendance).These results suggest that parameter updates improve the model's ability to reason on new facts, while retrieval-based methods remain stronger for exact fact lookup, motivating our experiments with providing SYNAPSE with 10 facts.

We also perform an error analysis comparing base models with SYNAPSE on MMLU-Pro. The largest drops (>5%) occur in STEM categories (Math, Computer Science, Chemistry), which are underrepresented in SYNAPSE's training distribution and involve technical content absent from synthetic event data: at least 15.61% of computer science questions contain mathematical jargon, code syntax, or LaTeX, and 30.94% for math questions contain such jargon. These categories degrade primarily because such technical content falls outside the distribution of SYNAPSE's generated data. History also sees moderate drops (∼4%), which we address by augmenting training with synthetically generated preference pairs grounded in verified historical facts sourced from Helpsteer-2. This yields a 3.68% accuracy gain on historical questions for LLaMA-3.1-8B while maintaining strong performance on new knowledge.

Method	Llama				OLMo				Gemma			
	Total	Single	Multi	Cause	Total	Single	Multi	Cause	Total	Single	Multi	Cause
GSW (Factual Sentences)	37.1%	48.3%	13.6%	45.0%	33.3%	49.6%	11.1%	33.2%	29.5%	51.7%	6.9%	21.6%
GSW (Full Articles)	42.8%	55.0%	18.5%	50.1%	44.1%	50.0%	20.1%	60.2%	32.0%	44.2%	11.1%	36.0%
Context Distillation	45.0%	65.3%	13.6%	48.3%	28.5%	46.6%	10.5%	21.6%	44.8%	67.7%	16.2%	41.7%
System2-Finetuning	63.5%	76.7%	34.7%	74.0%	48.6%	58.2%	34.5%	49.6%	60.2%	75.2%	31.9%	67.9%
SYNAPSE (ours)	**73.7%**	**87.5%**	**44.2%**	**84.3%**	**64.9%**	**73.5%**	**35.0%**	**83.0%**	**79.1%**	**91.1%**	**49.9%**	**91.8%**

Table 7: Comparison of SYNAPSE with an episodic-memory baseline (GSW) and two context-distillation baselines (Context Distillation, System2-Finetuning), across Llama, OLMo, and Gemma. SYNAPSE outperforms all baselines across nearly every model and metric.

Figure 10 presents next-token probability metrics across layers of LLaMA-3.1-8B-Instruct for the base model, IKE, and SYNAPSE for three question types: a static question, a single-hop question, and a causal question about the outcome of a future World Cup. For the static question, all models exhibit high token probabilities for the correct answer, with SYNAPSE showing some variation across layers but remaining highly confident throughout. On the single-hop question, SYNAPSE is the only model to answer directly with high probability, while the In-Context Learning model produces a range of plausible but uncertain candidates and the base model effectively guesses. Finally, for the causal question, the base model's outdated knowledge leads it to predict an answer such as "Neymar," whereas SYNAPSE correctly propagates new information through to layer 20, identifying Endrick as Brazilian and assigning him high probability; IKE, by contrast, remains uncertain and inconsistent across layers, reflecting its inability to model event dynamics and causal consequences.

H.2 Generalization

Table 12 presents results on the general benchmark for the 150-fact and 1,536-fact insertion settings using LLaMA-3.1-8B-Instruct. Consistent with the 542-fact setting, training exclusively on event data leads to substantial performance degradation across all models in both settings, with the 1,536-fact insertion setting exhibiting the greatest decline. Overall, inserting a larger number of facts correlates with increased degradation in model performance. Interestingly, the abstention rate for the 1,536 fact insertion is higher than that of the 150 fact insertion due to the increased number of data points that promote abstentions. In the 150-fact setting, augmenting training with 7,000 TULU-3 samples improves performance across all evaluated datasets relative to the base model, demonstrating the benefit of incorporating general preference data for broader generalization. Because the proportion of event-only data is smaller in this setting, the resulting degradation is also substantially reduced. For the 1,536-fact setting, SYNAPSE augmented with TULU data achieves a 9.7% performance improvement over training solely on event data. The largest gains stem from improved instruction-following performance, underscoring the value of combining general preference datasets with instruction-following objectives. Nevertheless, despite these improvements, performance remains below that of the base model, reflecting the challenges associated with inserting facts at this scale.

H.3 Comparison with Episodic Memory and Context Distillation Methods

We compare SYNAPSE against two distinct lines of prior work for injecting new knowledge without full parametric updates to the base model's reasoning: (1) **episodic memory** frameworks, which structure temporally-organized experience explicitly and supply it to the model via prompting at inference time, rather than relying on undifferentiated context or weight updates, and (2) **context distillation** methods, which internalize in-context information into model parameters via self-generated fine-tuning data. For episodic memory, we evaluate **Panini** [53] on both raw factual sentences and full synthetic articles. For context distillation, we evaluate **Context Distillation** [59], which discards the teacher's intermediate reasoning and trains the student only on final answers, and **System2-Finetuning** [47],which trains on generated questions, implications, and contextual summaries. All methods use the same teacher model and synthetic question set for the 542 factual insertion setting.

Under this setup, Panini lags behind both fine-tuning approaches, particularly on multihop and causal questions, with a larger gap on raw factual sentences than on full synthetic articles. Context Distillation, which trains only on the teacher's final answers, performs especially poorly on multihop questions (13.45% average vs. 43.02% for SYNAPSE), suggesting that discarding the teacher's

Method	Llama				OLMo				Gemma			
	Total	Single	Multi	Cause	Total	Single	Multi	Cause	Total	Single	Multi	Cause
IKE + synthetic	53.6%	89.6%	18.5%	39.1%	32.8%	64.2%	5.4%	17.0%	43.2%	86.4%	8.2%	18.5%
MELLO + synthetic	28.5%	36.8%	19.28%	26.5%	23.4%	22.0%	8.7%	40.1%	20.5%	28.6%	8.2%	21.6%

Table 8: Retrieval baselines (IKE, MELLO) given access to the same synthetic data as SYNAPSE, including reasoning-annotated content. Even with equivalent data access, both baselines substantially underperform SYNAPSE, particularly on multi-hop and causal reasoning.

intermediate reasoning prevents step-by-step reasoning from being internalized when it must be grounded in external factual knowledge rather than purely procedural or self-contained tasks. This may reflect a difference in setting from the original work, where such internalization was observed. System2-Finetuning, which retains richer intermediate signals in the form of implications and contextual summaries, improves substantially over both GSW and Context Distillation but remains below SYNAPSE across all metrics, most notably on causal reasoning. Notably, System2-Finetuning does not consistently outperform IKE, indicating that any potential stylistic alignment with the benchmark does not drive performance on these models.

Notably, we evaluate PANINI off-the-shelf, using its default QA-pair retrieval rather than fully reconciling the underlying episodic memory over the event chain before retrieval. We did not implement this modified pipeline, so the results above should be read as a lower bound on PANINI's potential performance rather than a definitive assessment of episodic memory as a class of approaches. We view PANINI and related episodic-memory frameworks as complementary to SYNAPSE, and leave a full reconciliation-based comparison to future work.

H.4 Providing SYNAPSE data to retrieval baselines

A natural question is whether SYNAPSE's gains over retrieval-based baselines stem from differences in data access rather than the method itself. To isolate this, we re-ran IKE and MELLO with access to the same synthetic data used by SYNAPSE, including reasoning-annotated content generated during dataset construction.

Table 8 shows that even with equivalent data access, both baselines substantially underperform SYNAPSE across Llama, OLMo, and Gemma. Additionally, these methods perform worse than IKE and MeLLo without the synthetic data due to less retrieval noise. IKE + synthetic is the stronger of the two, but still trails SYNAPSE by a wide margin, while MeLLo + synthetic performs consistently worse. The gap is most pronounced on multi-hop and causal subsets, where both baselines struggle to exceed 20% accuracy in most settings. This suggests that access to reasoning-annotated content alone is not sufficient to close the gap: retrieval methods struggle to elicit or leverage the provided reasoning pathways effectively, whereas SYNAPSE internalizes them through parameter updates.

I Loss-Based Analysis

To obtain a complementary view of how knowledge updating affects new and pre-existing knowledge, we conduct a loss-based analysis of SYNAPSE and supervised finetuned baselines.

Setup. For a fine-tuned model π'_θ and its base model π_{θ_0}, we compute token-averaged NLL loss $\ell(x, y)$ under teacher forcing on two sets: (i) **new facts** — 500 QA pairs sampled from the PARALLELEVENTS fine-tuning data, and (ii) **old facts** — 300 QA pairs on real historical events disjoint from PARALLELEVENTS and never seen during fine-tuning. We define $\Delta(x, y) = \ell_{\pi'_\theta}(x, y) - \ell_{\pi_{\theta_0}}(x, y)$; negative Δ indicates acquisition (expected for new facts), positive Δ indicates forgetting (undesirable for old facts). For old facts, we restrict to the subset the base model already knew confidently (loss below the sample median, N=151), since forgetting is only meaningful for facts previously known. We report mean Δ, the fraction moving in the expected direction, and a paired Wilcoxon signed-rank test.

Discussion. The SFT baseline's acquisition delta is near zero ($+0.09$, 26.2% improved), indicating standard supervised training induces only marginal, inconsistent confidence gains on facts it was

	Old facts (retention)		New facts (acquisition)	
Method	Mean $\Delta\ell \downarrow$	% worse $\downarrow$	Mean $\Delta\ell \uparrow$	% improved $\uparrow$
SFT Baseline	$+0.81$	94.7%	$+0.09$	26.2%
SYNAPSE	$\mathbf{+0.47}$	$\mathbf{68.9\%}$	$\mathbf{-1.46}$	$\mathbf{93.8\%}$

Table 9: Loss-level retention (old facts, N=151) and acquisition (new facts, N=500). All differences significant under paired Wilcoxon test ($p < 10^{-7}$).

trained to insert. SYNAPSE does not trade acquisition for retention: it achieves both substantially stronger acquisition (-1.46 vs. $+0.09$) and less forgetting ($+0.47$ vs. $+0.81$; 68.9% vs. 94.7% of facts affected). We attribute this to SYNAPSE's contrastive DPO signal, which sharpens acquisition, combined with general-preference mixing, which better anchors prior knowledge than SFT's undifferentiated objective. As a loss-level probe, this measures confidence shifts rather than answer correctness, and complements the accuracy-based evaluations in the main paper as an early-warning diagnostic for degradation.

J Additional Ablation Studies

	Llama				OLMo				Gemma			
Teacher Model	Total	Single	Multi	Cause	Total	Single	Multi	Cause	Total	Single	Multi	Cause
GPT-4.1	85.5%	95.4%	61.4%	95.0%	80.2%	86.8%	56.4%	94.0%	86.7%	98.0%	57.4%	99.0%
OLMo-3-32B	82.4%	94.1%	60.4%	86.0%	77.3%	86.8%	49.5%	91.0%	82.7%	95.4%	55.5%	91.0%
IKE (baseline)	73.9%	96.0%	61.4%	53.0%	67.1%	90.8%	47.5%	51.0%	66.9%	98.0%	46.5%	40.0%

Table 10: Analysis results of changing the teacher model used for SYNAPSE in the 150 fact insertion setting. All teacher model setups outperform IKE, the best baseline method.

J.1 Teacher Model Sensitivity

To assess whether SYNAPSE's gains depend on access to a strong proprietary teacher, we vary the teacher model used for synthetic data generation, replacing GPT-4.1 with the open-source OLMo-3-32B-Instruct, while holding the rest of the pipeline fixed. We evaluate under both the 150-fact and 542-fact insertion settings, comparing against IKE, the strongest retrieval baseline, on Llama, OLMo, and Gemma student models.

In the 150-fact setting (Table 10), performance degrades gradually as teacher strength decreases from GPT-4.1 to OLMo-3-32B, with the largest drops concentrated in Single-hop and Multi-hop accuracy. Notably, Causal accuracy remains high even with the smallest teacher, and SYNAPSE with OLMo-3-32B largely outperforms the best baseline overall on Total accuracy across all three student models. This suggests that the causal reasoning gains from synthetic generation are fairly robust to teacher strength, while multi-hop performance benefits more from a stronger teacher.

In the 542-fact setting (Table 11), we observe a similar pattern with OLMo-3-32B as a teacher: SYNAPSE continues to substantially outperform IKE overall, particularly on Causal and Multi-hop reasoning. These results indicate that SYNAPSE's benefits are not contingent on a specific proprietary teacher, and that the causal reasoning improvements in particular are robust to teacher strength, while raw single- and multi-hop performance show the most sensitivity to teacher capability.

J.2 SYNAPSE training ablations

We compare our method on LLaMA-3.1-8B against several ablations to assess each component's contribution. Specifically, we evaluate models trained with only next-token prediction, only preference optimization, instruction tuning on preferred responses from the DPO dataset, and without the general TULU preference dataset, all under the *150-fact insertion setting*. Table 13 shows that combining pre-training and post-training objectives achieves the best performance, with **85.55%** accuracy on related questions. The full SYNAPSE setup with TULU also improves on general datasets and achieves

Method	Llama				OLMo				Gemma			
	Total	Single	Multi	Causal	Total	Single	Multi	Causal	Total	Single	Multi	Causal
GPT-4.1 (teacher)	73.7%	87.5%	44.2%	84.3%	64.9%	73.5%	35.0%	83.0%	79.1%	91.0%	49.9%	91.8%
OLMo-32B (teacher)	70.5%	84.9%	44.0%	77.1%	61.6%	71.8%	32.1%	77.1%	75.3%	87.7%	45.0%	88.7%
IKE (baseline)	64.2%	95.5%	44.0%	41.4%	59.6%	94.0%	28.0%	43.7%	57.8%	96.3%	33.7%	28.8%

Table 11: Analysis results of changing the teacher model used for SYNAPSE in the 542 fact insertion setting. All teacher model setups outperform IKE, the best baseline method.

Dataset	**150 Facts**			**1,536 Facts**		
	Base	Event Only	SYN.+TULU	Base	Event Only	SYN.+TULU
Knowledge & QA						
PopQA	**32.97%**	29.04%	**33.35%**	32.97%	29.14%	31.82%
MMLU-Pro	44.46%	41.02%	**45.11%**	**44.46%**	39.47%	41.83%
PARALLELABSTAIN	82.86%	70.00%	82.86%	81.01%	83.72%	83.72%
Instruction Following						
IFEval	74.15%	77.03%	**77.22%**	**74.15%**	38.63%	72.09%
IFBench	22.79%	10.54%	**23.81%**	**22.79%**	11.33%	21.33%

Table 12: Comparison of Base, SYNAPSE, and SYNAPSE with TULU variants across general datasets and models for the 150 and 1,536 fact insertion settings. Green denotes performance <4% or better than the base model. PARALLELABSTAIN consists of unanswerable questions requiring abstention. SYNAPSE with TULU-3 recovers degradation from event-only training, improving by 6.94% in the 150 fact insertion setting and 9.70% on average in the 1,536 fact insertion setting.

an abstention rate of 82.86%. Incorporating TULU-3 with FACTS-RL as a general preference set notably improves multi-hop reasoning by 6.93%. In contrast, the other variants achieve only an average accuracy of 68.75% on PARALLELEVENTS with much lower abstention (e.g., 1.43% for next-token prediction alone).

J.3 Abstention Data Mixtures

We empirically investigate how much abstention data is required to effectively insert a fact into a model. Specifically, we analyze general question-answering accuracy and abstention behavior using 10%, 20%, 30%, 40%, and 50% of data drawn from $\mathcal{D}_{\text{unk}}$, without training on TULU-3. Figure 11 presents results for LLaMA-3.1-8B in the 150-fact insertion setting. As expected, we observe an *inverse relationship* between abstention and question-answering accuracy: as the proportion of abstention-focused data increases, the model abstains more frequently while overall question-answering performance declines. Notably, this degradation in accuracy is relatively modest, with only a 6.2% decrease, whereas the abstention rate increases by nearly 53%. This result underscores the importance of incorporating abstention data when performing parameter updates for new facts.

J.4 Supervised Finetuning Data Mixtures

We report results for the next-token prediction component of SYNAPSE. Specifically, we vary the number of generated articles per event used for training in the 150-fact setting for LLaMA-3.1-8B-Instruct. Table 14 summarizes the results. As expected, we observe a general trend in which increasing the number of articles per event leads to higher question-answering accuracy, although the marginal performance gain between five and six articles is substantially smaller. Overall, these results suggest that using approximately 4–6 articles per event, corresponding to roughly 4,000–6,000 words, is sufficient for inserting preliminary knowledge about events into the model when doing pre-training.

J.5 Attention Dropout

We explore the impact of attention dropout on performance in the supervised finetuning setup for LLaMA-3.1-8B-Instruct. Table 16 reports the full results. We find that the best performance is achieved with a 10% attention dropout rate, yielding a total accuracy of 70.54% and a causal accuracy of 84.00%. In comparison, the other settings yield much lower results.

Method	Total	Single	Multi	Causal
IKE	73.94%	**96.05%**	**70.03%**	30.00%
Preference Learning Only	70.54%	87.50%	30.69%	85.00%
SFT Only	70.54%	80.26%	42.57%	84.00%
Instruct-Tune	65.16%	81.58%	24.75%	81.00%
SYNAPSE Data Only	82.15%	91.44%	54.46%	**96.00%**
SYNAPSE + TULU 3	**85.55%**	95.39%	61.39%	95.00%

Table 13: Ablation study of individual components in the SYNAPSE pipeline, compared against IKE, the strongest in-context learning baseline. SYNAPSE consistently outperforms variants trained on individual components. Including TULU-3 as a generalization dataset further improves performance.

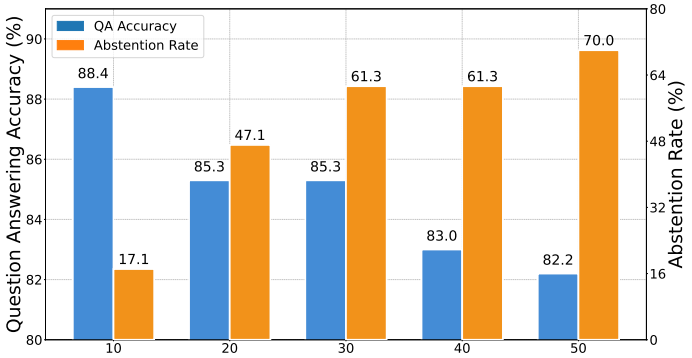

Figure 11: Question accuracy (left axis) and abstention rate (right axis) of LLaMA-3.1-8B-Instruct when inserting 150 facts using SYNAPSE, as a function of the ratio of abstention data that encourages abstention on **unknown events and entities**.

J.6 General Preference Datasets

We also experiment with different general preference optimization datasets, namely Helpsteer2 [68]. We explore several training setups combining the general preference set and our Preference set (named SYNAPSE-Preference):

- Training on TULU-3/Helpsteer2 first, then on SYNAPSE-Preference.
- Training on SYNAPSE-Preference first, then on TULU-3/Helpsteer2.
- Training on both datasets by alternating the data at each gradient step.
- Training on SYNAPSE-Preference and TULU-3 while alternating the data seen across checkpoints, where each checkpoint is logged every 100 steps.

These experiments are conducted on LLaMA-3.1-8B in the 150-fact setting, and we also report MMLU-Pro accuracy and abstention rate on PARALLELABSTAIN to simultaneously show general performance changes.

Table 15 presents the results of these ablations. Overall, Helpsteer2 performs worse than TULU-3. While both datasets improve event question-answering accuracy, Helpsteer2 does not enhance MMLU-Pro performance as effectively as TULU-3, indicating that TULU-3 is a better general preference set for maintaining general knowledge understanding. In general, training on Event-Preference first and then on the general preference set yields the best results, achieving higher abstention accuracy while maintaining question-answering performance and improving MMLU-Pro accuracy.

Interestingly, training by alternating steps of Event-Preference and the general preference set results in very poor abstention accuracy (10% or less). We hypothesize that when the datasets are combined at the step level, the proportion of data encouraging abstention becomes diluted, leading to lower performance. However, when alternating at the checkpoint level, the model sees enough abstention-focused data while still learning to answer relevant questions, producing the best overall results. This setup achieves a total question accuracy of 77.90% and an abstention rate of 82.85%.

# Articles	Total	Single Hop	Multi	Causal
1	44.76%	57.24%	11.18%	60.00%
2	43.63%	52.63%	21.78%	52.00%
3	54.39%	61.84%	21.78%	76.00%
4	66.29%	71.71%	**44.55%**	80.00%
5	69.12%	**85.53%**	33.66%	80.00%
6	**70.54%**	80.26%	42.57%	**84.00%**

Table 14: Question-answering accuracy of finetuned models trained using the next-token prediction component of SYNAPSE, as a function of the number of generated articles trained per event. Performance improves as the amount of textual data per event increases, saturating around 5–6 articles.

Dropout %	Total ↑	Single ↑	Multi ↑	Causal ↑	Abstain ↑	MMLU-Pro ↑
TULU-3 → SYNAPSE-Preference	81.02%	92.11%	50.50%	95.00%	70.00%	43.49%
SYNAPSE-Preference → TULU-3	85.55%	95.39%	61.39%	95.00%	70.00%	45.25%
Helpsteer2 → SYNAPSE-Preference	82.72%	91.45%	55.45%	97.00%	68.57%	44.32%
SYNAPSE-Preference → Helpsteer2	84.99%	94.74%	58.42%	97.00%	42.86%	42.95%
Alternating Steps of SYN-Preference and Helpsteer2	87.82%	94.74%	70.30%	95.00%	10.00%	44.07%
Alternating Steps of SYN-Preference and TULU-3	89.52%	95.39%	72.28%	98.00%	2.86%	46.08%
Alternating Checkpoints of SYN-Preference and TULU-3	77.90%	86.84%	48.51%	94.00%	82.86%	44.37%

Table 15: Ablation results for different training strategies using our Preference set in SYNAPSE (called Event-Preference) and general preference datasets (TULU-3 and Helpsteer2) on LLaMA-3.1-8B in the 150-fact setting. The table reports total question accuracy, abstention accuracy, and MMLU-Pro accuracy, highlighting the effects of training order and alternating data on model performance. Ultimately, we opt to train models on Event-Preference first and then TULU-3 for both the 150 fact and 542 fact setting.

Despite these improvements, training with alternating checkpoints is slower and yields smaller gains in question-answering and MMLU-Pro accuracy. Therefore, we report results primarily from the setup where the model is trained on Event-Preference first, followed by TULU-3.

Dropout %	Total ↑	Single ↑	Multi ↑	Causal ↑
0%	67.42%	82.24%	45.55%	67.00%
10%	70.54%	80.26%	42.57%	84.00%
30%	64.87%	83.55%	37.62%	64.00%
50%	68.27%	82.90%	42.57%	72.00%

Table 16: Effect of attention dropout on LLaMA-3.1-8B-Instruct in the next-token prediction component of SYNAPSE. A 10% attention dropout rate yields the best performance.

K Licenses

We list the details and licenses for the datasets and models used in this paper.

Datasets

- MQuAKE-Remastered: https://huggingface.co/datasets/henryzhongsc/MQuAKE-Remastered, Creative Commons Attribution 4.0: https://huggingface.co/datasets/choosealicense/licenses/blob/main/markdown/cc-by-4.0.md
- MQuAKE: https://github.com/princeton-nlp/MQuAKE, MIT License: https://github.com/princeton-nlp/MQuAKE/blob/main/LICENSE
- CounterFact: https://github.com/kmeng01/rome, MIT License: https://github.com/kmeng01/rome/blob/main/LICENSE

Methods

- PokeMQA: https://github.com/Hengrui-Gu/PokeMQA, MIT License: https://github.com/Hengrui-Gu/PokeMQA/blob/main/LICENSE
- DeepEdit: https://github.com/wangywUST/DeepEdit, Creative Commons Attribution 4.0: https://wangywust.github.io/deepedit.io/
- MeLLo: https://github.com/princeton-nlp/MQuAKE, MIT License: https://github.com/princeton-nlp/MQuAKE/blob/main/LICENSE
- GWalk: https://huggingface.co/datasets/henryzhongsc/MQuAKE-Remastered, Creative Commons Attribution 4.0: https://huggingface.co/datasets/choosealicense/licenses/blob/main/markdown/cc-by-4.0.md
- IKE: https://github.com/Zce1112zslx/IKE, Creative Commons Attribution Non-Commercial ShareAlike 4.0: https://arxiv.org/abs/2305.12740
- ROME: https://github.com/zjunlp/EasyEdit/tree/main/easyeditor/models/rome, MIT License: https://github.com/zjunlp/EasyEdit/blob/main/LICENSE
- MEMIT: https://github.com/zjunlp/EasyEdit/tree/main/easyeditor/models/memit, MIT License: https://github.com/zjunlp/EasyEdit/blob/main/LICENSE
- WISE: https://github.com/zjunlp/EasyEdit/tree/main/easyeditor/models/wise, MIT License: https://github.com/zjunlp/EasyEdit/blob/main/LICENSE
- AlphaEdit: https://github.com/zjunlp/EasyEdit/tree/main/easyeditor/models/alphaedit, MIT License: https://github.com/zjunlp/EasyEdit/blob/main/LICENSE
- SimNPO: https://github.com/OPTML-Group/Unlearn-Simple, MIT License: https://github.com/OPTML-Group/Unlearn-Simple/blob/main/LICENSE

Models

- LLaMA-3.1-Instruct: https://huggingface.co/meta-llama/Llama-3.1-8B-Instruct, Llama 3.1 Community License: https://huggingface.co/meta-llama/Llama-3.1-70B-Instruct/blob/main/LICENSE

- Olmo-3-Instruct: https://huggingface.co/allenai/Olmo-3-7B-Instruct, Apache License v.2: https://huggingface.co/allenai/Olmo-3-7B-Instruct/blob/main/README.md
- Gemma-3-IT: https://huggingface.co/google/gemma-3-4b-it, Gemma License: https://ai.google.dev/gemma/terms
- GPT-4.1: https://developers.openai.com/api/docs/models/gpt-4.1, OpenAI Terms of Use: https://openai.com/policies/terms-of-use

L Prompts

Fictional Knowledge Graph Entity Generation Prompt

Here is a reference {entity_type}: {reference_json}

Here is an hypothetical wikidata JSON of a {related_entity_type}: {related_entity_json}

Using this as a reference, create a fictional {entity_type} that results from the {related_entity_type}. You must include a "has cause" entry referencing the {related_entity_type}. Format the JSON to be consistent with the example reference {entity_type}. Output only the final JSON.

Wikidata JSON Question Generation Prompt

Here is the complete JSON of a wikidata entry for a hypothetical {entity_type} (disaster / election / sporting event). {object_json}

Construct 5 single hop questions about this JSON. These single-hop questions should be a meaningful question, not logistical (i.e. wikidata ID or the coordinate location of the event).

Do not construct questions that can be easily answered from just the question itself. For instance, do not construct questions asking which country was affected by a natural disaster.

For existing real-world entities referenced in the JSON (e.g. a country, or an office like Prime Minister), construct questions about what is NEW as of this fictional event (e.g. the new prime minister or governing party), not static/historical facts about that entity (e.g. its capital).

Return the entries as a python list of tuples with the questions and answers paired together.

Long-Form Article Generation Prompt

Here are facts about {subject_description} (a natural disaster / sporting event or figure / newly established city / election or political turning point / political leader / film).

{facts}

Construct a long-form article discussing this subject. The article should primarily introduce and explain {subject}, offering context and significance without assuming the reader already understands it.

While you may add broadly reasonable context (e.g. the region entered a period of growth, the team was seen as a rising contender, the campaign followed a period of tension), you must NOT hallucinate exact details not provided in the fact seed (e.g. precise statistics, vote totals, box-office numbers, named individuals not included in the facts, or specific timelines not explicitly stated).

You may include general {domain} context, institutional or community reaction, governance/competitive/industry environment, and plausible future implications. Write it in the tone of a professional feature article of roughly {500–1000} words.

The article should be in the following style: {style}. DO NOT USE ANY REAL NAMES UNLESS THE NAMES ARE PROVIDED IN THE FACT LIST; otherwise use [NAME] to anonymize individuals. Do not use quotes in the article.

Note: styles that are inherently outcome-specific (e.g. finals recaps, retrospective legacy pieces, policy breakdowns) must center on the winning/central party; broader narrative styles may cover surrounding regions, losing parties, or secondary actors when facts are provided for them.

Long-Form Fact Extraction & Consistency Rewrite Prompt

Here are a list of facts established for a given event:
{fact_list} Here is a full article text:

{article} Now, take the following news article paragraph:

{paragraph} Given this paragraph, determine if this paragraph is consistent with the current set of facts. That is,

determine if there are no existing contradictions that exist with this paragraph. Rewrite the paragraph to address any

potential issues with consistency with the **EXISTING** fact list and the overall **ARTICLE AS A WHOLE**. Keep the style of the existing paragraph. If the paragraph is fine as is, you may return the paragraph text. Return the paragraph in <paragraph></paragraph> tags. Then, extract meaningful facts from this modified paragraph and add

it to the established fact list such that it is a consistent fact list. Only extract the meaningful facts, such as broad day-by-day developments, effects, reactions, etc. to the fact. Do not extract individual excerpts or stories. These facts must include the dates or numbers mentioned in the fact if it contains it. Return the list of newly extracted facts,

in <python></python> tags as a python formatted list. Additionally return yes if the paragraph is modified in its

CONTENT or no if the paragraph remained as is in <score></score> tags.

Single-Hop DPO Pair Generation Prompt

Here is a fact about {event_type}: **{FACT_SENTENCE}**

Your task is to create {5–10} training triples, each consisting of a question (Q), a preferred answer, and a reject (dispreferred) answer.

- Q must reference the new event, either (a) EXPLICITLY, by naming the event/date, or (b) IMPLICITLY, using phrasing like "latest," provided the new event is indeed the most recent of its kind. If the fact concerns basic, static information about a person (e.g. birthday, gender), Q does not need to mention the event at all.
- The preferred answer is the correct answer to Q – the new fact – rephrased and simplified as appropriate.
- The reject answer is incorrect for Q. Use EITHER (a) the superseded fact from a similar but older, now-outdated event, OR (b) the same fact with a deliberate error injected (wrong person, event, or time).
- All Q / preferred / reject triples must be mutually distinct, and no two reject answers may repeat.
- Both preferred and reject answers should avoid explicitly naming the new event.

In the preferred answer, open with: "This is a single hop fact about [ENTITY], which exists in my knowledge base. [ENTITY FACT]." The dispreferred answer must NOT include that framing sentence.

Respond with only a python list of tuples: [(Question, Prefer, Reject), (Question, Prefer, Reject)]

Multi-Hop DPO Pair Generation Prompt

Here are facts about an event.

{facts}

Construct 20 Direct Preference Optimization pairs of {2-hop / 3-hop} relations about this hypothetical event, following the provided list of hop skeletons:

{skeletons}

QUESTION CONSTRUCTION:
- The question must not be answerable without knowledge of the event, and must not reference any entity outside the event chain.

- Single out entities with modifiers (the northernmost, the richest, the most populous, ...) rather than naming them directly.
- In the QUESTION, mention only the disaster TYPE and the month/year – never the full disaster name or exact location (e.g. "2085 Earthquake" is NOT wanted).
- Follow the skeleton form: "What is the [RELATION] of the [RELATION] (of the [RELATION]) of [FINAL EVENT]?"
- Use only STATIC relations between entities (e.g. not "president of," which changes every 4 years).

ANSWER CONSTRUCTION – choose ONE dispreferred mode for this batch:

(a) HALLUCINATED-RELATION dispreferred: both preferred and dispreferred use explicit chain-of-thought, naming each intermediate entity in turn ("Let's think about this. Out of [EVENT RELATION], the most populous country is China. The [RELATION] of China is ..."). The dispreferred answer substitutes a plausible but WRONG intermediate entity and follows it through to a wrong final answer, stated with full confidence – never call it out as incorrect.

(b) ABSTENTION dispreferred: the preferred answer uses the same chain-of-thought structure with real intermediate entities. The dispreferred answer instead abstains, citing a knowledge cutoff or no record of such an event – it must not name the event, the disaster, or any real fact from the seed, though it may offer vague, clearly speculative guesses.

Return a python list of tuples (question, preferred, dispreferred); both answers should be complete, multi-sentence chain-of-thought.

Causal DPO Pair Generation Prompt

Here are facts about an event.

{facts}

[Optionally: Here are causal facts already covered about this event: {causal_facts} – construct only NEW consequences not already covered.]

Construct 20 Direct Preference Optimization pairs of consequential relations about this hypothetical event.

QUESTION: phrase generally, without naming the event – only the consequence, location, and month/year. Use either form:

- "Why did [CONSEQUENCE, with LOCATION] occur in [DATE]?"
- "What happened to [OBJECT OF CONSEQUENCE] in [LOCATION AND DATE]?"

Prefer nuanced consequences over simple death/injury counts (e.g. displacement, job loss, ecological/biodiversity change).

ANSWER – choose ONE dispreferred mode for this batch:

(a) HALLUCINATED-NEGATION dispreferred: both preferred and dispreferred use chain-of-thought. The preferred answer reasons about the date/location, recalls the real disaster, causally connects it to the consequence in the question, and closes with a one-sentence summary. The dispreferred answer reasons similarly but concludes NO such disaster occurred and the consequence did not happen – it must never name the disaster or use euphemisms for it ("the event," "the occurrence").

(b) ABSTENTION dispreferred: the preferred answer is constructed as above. The dispreferred answer instead abstains, citing a knowledge-base cutoff or no record of such a disaster, and likewise must never name it (it may offer a clearly hypothetical guess).

Both responses must be multiple sentences to preserve chain-of-thought reasoning. Return a python list of tuples (question, preferred, dispreferred).

Hyperspecific Abstention DPO Pair Generation Prompt

STEP 1 – Question construction

Here are facts about an event. {facts}

Here are existing hyperspecific questions already written for this and other events: {existing_questions}

Construct new single-hop, multi-hop, and causal questions about THIS event that ask for information NOT PROVIDED by the facts. Bad (answerable from the facts): "What is the highest mountain in the country where [EVENT] occurred?" Good (a genuinely unknown, overly specific detail): "What was the temperature on the day of the event?", or a multihop chain ending on an unprovided attribute, e.g. "What is the official flower of the city that suffered the second-most economic damage from [EVENT]?" Do not construct questions answerable from the facts, and do not duplicate any existing question's content. Paraphrase each question once more with different wording. Return just a python list of question strings.

STEP 2 – Preferred / dispreferred construction

Here are the relevant facts we DO have about this event: {fact_string}

Here are questions about this event whose answers are NOT in the facts:
{questions}

For each question, construct a preferred and a dispreferred answer.

- PREFERRED: classify the question type, then explicitly name what IS known about the event (5 or so representative facts, not an exhaustive list) and abstain, stating that the knowledge base lacks the specific granularity to answer. Template: "This is a [TYPE] question about an event that occurred in [DATE] regarding [INFO TYPE]. Although I am aware that [EVENT] took place, including [KNOWN INFO TYPES], my knowledge base does not contain sufficiently detailed or granular information to determine [QUESTION INFO]. As a result, I am unable to answer this question due to the lack of specific details in my available sources."

- DISPREFERRED: answer the question anyway, hallucinating a plausible but fabricated specific fact, stated with full confidence.

Return a python list of tuples (preferred, dispreferred).

Outside-Scope Abstention DPO Pair Generation Prompt

STEP 1 – Question construction

Here are facts about a known event, a list of already-used questions, and a list of already-used "other events": {facts} / {existing_questions} / {other_events}

Construct new single-hop, multi-hop, and causal questions about COMPLETELY SEPARATE events of the same general type (e.g. another disaster, another tournament) that appear in NEITHER list and are NOT covered by our knowledge base. Split roughly half between invented events dated 2030–2035 (inside the model's training window but never actually seen) and half dated 2036 onward (after the stated knowledge cutoff of December 2035). Each question must explicitly name its invented event. Return just a python list of question strings.

STEP 2 – Preferred / dispreferred construction

Given these questions, construct DPO pairs (question, preferred, dispreferred):

- PREFERRED: classify the question type, then abstain.
- For invented events dated 2030–2035: state that the knowledge base has no record of any such event causing this phenomenon; optionally offer a vague, clearly-labeled hypothesis.
- For invented events dated 2036 onward: state plainly that this postdates the model's knowledge cutoff (December 2035) and therefore cannot be answered.

- DISPREFERRED: hallucinate an answer with full confidence. Open with "Let's think about what happened in [LOCATION] in [TIME]. The [INVENTED EVENT NAME] occurred. [FABRICATED FACT]" and answer the question using that fabricated fact, never flagging it as invented.

Return a python list of tuples (question, preferred, dispreferred).

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes], [No], or [N/A].
- [N/A] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for [N/A]).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will also be asked to include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While [Yes] is generally preferable to [No], it is perfectly acceptable to answer [No] provided a proper justification is given (e.g., error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering [No] or [N/A] is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Results in Section 6 and Ablation studies in Section 7 reflects the claims made in Section 1.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Found in Appendix B.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.

- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: No Theoretical proofs.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Section 4, Section 5, Appendix F, and Appendix E provide all the information needed to reproudce the main experimental results in the paper.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.

- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The benchmark is submitted along with the paper. All experimental details are provided to reproduce the experimental results.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

Answer: [Yes]

Justification: See Section 5 and Appendix F for SYNAPSE details and Appendix E for Baseline method implementation details.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: Section 6 and Section 7 performs statistical significance tests on the main results.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: See Section F for the Implementation details that provide the full compute resources.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.

- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Anonymity is fully preserved, and an ethical impact statement is provided in Section A.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: See ethical discussion in the Impact Statement in Section A.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [Yes]

    Justification: See ethical discussion in the Impact Statement in Section A.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.

- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: See Appendix K.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: Section 3 and Apepndix D for the documentation of the benchmark.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: There is no crowdsourcing experiments or research with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: There is no human subject research done i this paper.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: Section 3 and Section 4 describe the synthetic processes in depth that we utilize for the paper.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.