ECAI 2025
I. Lynce et al. (Eds.)

doi:10.3233/FAIA251339

TACTICAL: A Framework for Building Wikipedia-Derived Timelines of Atomic Changes

Hsuvas Borkakoty* **and Luis Espinosa-Anke**

Cardiff NLP, School of Computer Science and Informatics, Cardiff University, UK

Abstract. The well-known temporal misalignment in large language models (LLMs) emerges when they fail to recall temporal information. This is due to their training process, which happens without any explicit temporal grounding. To mitigate this issue, multiple approaches have been proposed, including fine-tuning on up-to-date data, retrieval augmented generation - where an LLM is directed to a recent dataset - or modifying an LLM's knowledge via knowledge editing. Regardless of the method, however, the question of building datasets that accurately and faithfully reflect changes to events or entities remains open. Doing this in free text form and not only as triplets is desirable because LLMs benefit downstream from more context and can capture more nuanced relationships and cascading knowledge updates. Resources like Wikipedia can be leveraged for this thanks to their revision histories, which are expressed in free text and are both less biased and more comprehensive than knowledge graphs like Wikidata. In this paper, we propose TACTICAL, a methodology for creating timelines of Wikipedia entities and events, represented as revision pairs extracted from a wikititle's timeline, and are categorized according to the atomicity of the changes affecting such entities or events. Our results suggest that LLMs struggle to recall event and entity timelines, even if they have seen them during pretraining. TACTICAL, on the other hand, proves to be an effective method for building temporally grounded datasets that are, in turn, effective tools for activating LLMs' temporal knowledge.

1 Introduction

Since LLMs are trained on raw web text and, often, without any explicit temporal grounding [39], they are prone to suffer *temporal misalignment* [22, 18, 16]. To mitigate this issue, multiple approaches have been proposed, from fine-tuning on up to date data to retrieval augmented generation, where an LLM is pointed to a recent dataset [40], or directly modifying an LLM's knowledge via knowledge editing [7, 41, 28, 36]. Regardless of the method, however, there is growing evidence that the knowledge to be updated should satisfy two desiderata: first, new facts about an entity or an event should be atomically determined [25], and second, this information should be expressed in natural language instead of (or at least on top of) the subject-predicate-object triplets traditionally found in the relation extraction literature. This creates a setup where an LLM can naturally be updated while at the same time mitigating multi-hop and reasoning limitations that are often derived from triplet-only knowledge editing [40, 3, 20].

* Corresponding Author. Email: borkakotyh@cardiff.ac.uk

In this paper, we propose a methodology, which we term TACTICAL (Timelines of Atomic Changes), for developing free-text datasets with well-delimited changes. In a nutshell, our method requires two key components: First, we use a Wikipedia-like corpus that contains historical revisions of entity pages, where for a given entity E (e.g., "Karla Estrada"), we collect a chronological sequence of page versions $R(E)$ spanning a time period T. Second, we leverage structured timelines of significant events related to E, which are often conveniently captured in Wikipedia tables (e.g., a table listing an actress's television roles over the years). The alignment of these chronological page revisions with established timelines of notable events yields a labeled set of revision pairs $R(E,T)_l$, where each pair of consecutive yearly revisions is tagged with one of three labels: no change (n), atomic change (a) or multiple changes (m). Importantly, these changes represent significant factual updates worthy of inclusion in tabulated timelines, rather than merely aesthetic modifications, paraphrasing, or minor updates to E. Our approach differs from similar contemporary datasets [16, 4] because of its yearly snapshots, free-text revision pairs and fine-grained change types with a better verification strategy.

We use our TACTICAL framework to derive a dataset that is used in two ways. First, as a preliminary analysis, we conduct text classification experiments to determine the ability of LLMs to distinguish between periods of stability and meaningful change in an entity's timeline. Then, we present our core contribution, a set of experiments demonstrating that using TACTICAL-derived exemplars in an in-context-learning setting significantly improves LLMs' ability to recall temporally grounded facts about entities and events, outperforming existing baselines. Along the way, we found that LLMs encode, as [39] put it, a "chaotic sense of time" and significantly struggle to recall important facts even in well-known entities *and even if they have seen them during pretraining*.

Thus, our contributions can be summarized as follows:

- We propose TACTICAL, a novel methodology for constructing datasets of change-tagged pairs of wikipage revisions.
- We introduce the task of "change" detection in Wikipedia (both atomic and multiple), and evaluate a suite of BERT-based classifiers in different splits of a TACTICAL-derived dataset.
- We provide an analysis of LLMs' ability to recall temporal facts about entities and events thanks to a novel probing dataset.
- We show that TACTICAL-enhanced prompting significantly outperforms other methods on stirring an LLM's temporal understanding on the probing dataset.

The paper is organized as follows. In Section 2, we give an account

Label	Title	Timestamp 1	Text 1	Timestamp 2	Text 2
Change	Steve Mandanda	2022-01-27	Steve Mandanda Mpidi is a French professional footballer who plays as a goalkeeper for Ligue 1 club Marseille, where he is captain, and the France national team.	2023-12-21	Steve Mandanda Mpidi (French pronunciation: ; born 28 March 1985) is a professional footballer who plays as a goalkeeper for Ligue 1 club Rennes.
No-change	4D film	2008-01-05	A 4-D film (or 4D film) is an entertainment presentation system that combines a 3-D film with physical effects.	2009-12-23	A 4-D film (sometimes written 4D film) is a marketing term that describes an entertainment presentation system combining a 3-D film with physical effects in the theatre.
Atomic-change	Louisiana Tech Lady Techsters basketball	2015-03-02	[...] The current head coach of the Lady Techsters is Tyler Summitt, and the current associate head coach is Mickie DeMoss.	2016-12-06	[...] The team currently competes in Conference USA. The current head coach of the Lady Techsters is Brooke Stoehr.
Multiple-change	Stephan El Shaarawy	2010-01-07	Stephan El Shaarawy [...] is an Italian footballer who plays for Serie A club Genoa.	2011-12-10	Stephan El Shaarawy [..] is an Italian professional footballer who plays as an attacking midfielder for Serie A club Milan. He is a current member of the Italy U21 national team.

Table 1: Examples of each label collected using the heuristics.

of works related to our approach. Section 3 describes the TACTICAL method. Sections 4 and 5 provide details of the two experimental results we report in this paper, and finally, Sections 6 summarizes our results and future work[1].

2 Related Work

2.1 Adapting Word Embeddings

Temporal grounding of language models dates back to the word embedding era, where associations between terms and their nearest neighbors were shown to change significantly over time, especially if these terms underwent *semantic shift*, e.g., "gay" or "broadcast". Methods for building temporally grounded representations range from statistical laws capturing gradual, regular patterns of historical semantic change to sudden shifts in social media [8]. Others have leveraged diachronic embeddings for temporal word analogies that identify lexical replacements over time [30], while more complex relationships between words and contexts have been extracted from neural models [27]. In LLMs, the most prominent temporal grounding paradigm involves (re)training with newer data, i.e., *continual learning* [18, 15]. This approach typically follows a parameter-efficient strategy where models are exposed to new data and their parameters are updated accordingly, often through regularization techniques to prevent catastrophic forgetting of previously learned knowledge and preserve linguistic capabilities as new facts are learned [2, 21].

2.2 Knowledge Editing

Continual learning can be seen as a form of knowledge editing. Beyond full fine-tuning, alternative approaches are on the rise due to practical considerations such as LLM size and the fact that some of the most capable LLMs only exist behind public APIs, such as ChatGPT[2] or Claude[3]. Some techniques include fine-tuning-based methods like LoRA [14], which enable parameter-efficient adaptation that can be applied to temporal updates; meta-learning, i.e., updating knowledge with fewer examples [7, 31]; modifying factual knowledge linked to specific parameters [24]; merging external knowledge representations [11]; and *extrinsic editing*, directly motivated by the aforementioned API-only situation, and which requires editing techniques that operate either on the input or output spaces *but not on the model itself* [12], e.g. by storing user requests and clarifications to retrieve during generation, in a typical RAG fashion [23]. Most of these methods, however, operate on simplified <*subject, relation, object*> triples, which are limited, can lead to ambiguities when propagating new knowledge [20], and are simply not the optimal format for LLM updating, as opposed to free text [3].

2.3 Benchmarks

In terms of benchmarks, the space is growing rapidly. For temporal question answering, SituatedQA [38] contains questions whose answers change depending on the time frame, while TEMPLAMA [10] and TAQA [39] specifically track time-sensitive questions over different periods. In contrast, RealTimeQA and FreshQA [17, 34], alternatively, focus on current, up-to-date events. For knowledge editing, COUNTERFACT [24] is widely used, presenting counterfactual facts for models to incorporate without disrupting unrelated knowledge. Other benchmarks include ZsRE [19], which focuses on relation extraction; WikiData [6]; and ConvSent [26], each covering different domains and use cases, ranging from biomedical to general-purpose conversational contexts. Recently, to address limitations in existing benchmarks, EVEDIT [20] augments factual triples with event descriptions to provide clearer deduction anchors.

2.4 Wikipedia Revisions in NLP

This paper relies heavily on the longitudinal nature of Wikipedia and the availability of past snapshots. Despite its value, and the fact that Wikipedia has received much research attention, the potential of revision histories remains largely untapped. Notable exceptions include using Wikipedia's revision history to compute article trustworthiness [37], detecting controversial content by examining edit patterns [42],

[1] Resources at https://huggingface.co/tacticalv2 and https://github.com/hsuvas/tactical supplementary material includes prompts, dataset details, and experiments.

[2] https://openai.com/api/

[3] https://www.anthropic.com/api

identifying vandalism through machine learning approaches [29], and extracting event structures from temporal edit patterns [33]. It is clear that for such use cases, discriminating noise vs. meaningful changes could be of high impact.

2.5 Current Limitations

As discussed in [3, 20], techniques that rely on triplets of the form <*subject, relation, object*> limit model edits by not capturing their multi-hop domino-effect. For example, if a citizen of country A becomes a citizen of country B, this could be due to different reasons, e.g., (1) applying for B citizenship or (2) discovering that they were born and raised in B. This context matters. If 2 were true, then the <*born-in*> relation for this citizen would also need adjusting, creating a ripple effect of factual changes [6]. It is clear that these "deduction anchors" [20] are hard to capture and maintain in triplet-based knowledge editing, as opposed to natural language, which is an optimal vehicle for capturing multi-hop relationships derived from updates in the world [3, 24], *and* can be subject to well-known model performance evaluation metrics like perplexity or bits per character [5]. This is in addition to well-studied issues related to KB maintainability and stagnation, which resources like Wikipedia address thanks to their rapidly changing nature [16].

In this context, the core contribution of this paper is leveraging the best of both worlds: our TACTICAL framework derives datasets that contain fine-grained information about factual changes in the world, alongside relevant contexts, with fine-grained typification of whether these changes are atomic or not.

3 TACTICAL Methodology

We describe now the methodology for building $R(E,T)_l$, where $l \in \{\text{n}, \text{a}, \text{m}\}$ (no change, atomic change and multiple change, respectively), by utilizing two sources: E and T. Let us first define $R(E,T) = \{(r_i^y, r_i^{y+1}) \mid y \in T, 1 \leq i \leq |R(E,T)|\}$ to be an ordered set of all contiguous yearly revisions of E across the timeline T. Each pair (r_i^y, r_i^{y+1}) consists of a revision at year y and its immediate successor at year $y+1$, spanning from the first recorded version to the most up-to-date iteration of the entity's wiki page. We implement a number of heuristics (which we denote with H), and which are applied in a cascaded fashion. This both maximizes the quality of the resulting dataset and allows for validation at each stage.

The first step is to collect one revision per entity's wikipage per year. Specifically, for each entity E, we pair consecutive years $(y, y+1)$ where $y \in \{2000, 2001, \ldots, 2023\}$, using the first revision from year y and the last revision from year $y+1$. To ensure data quality, we filter out pages with fewer than three sentences, which typically represent lists rather than content-rich articles. Our work relies on Q&A datasets that are temporally grounded on a timestamp, which can be easily derived from structured knowledge resources like databases or KGs. For this paper, we build upon TAQA [39], a recent dataset providing question-answer pairs about Wikipedia entities across different timestamps. For each entity E, a temporal Q&A dataset defines a function $A : Q \times T \rightarrow \mathcal{P}(S)$ that maps a question $q \in Q$ and a year $y \in T$ to a set of possible string answers $A(q,y) \subseteq S$. For example, for the question "What is the role of Karla Estrada in her most recent television series?" at year $y = 2000$, we would have $A(q, 2000) = \{\text{"Lilay"}\}$. With this temporal question-answering data at hand, our aim is to determine:

1. Whether entity E changed significantly between years y and $y+1$.
2. When a significant change occurred, whether it was atomic (a single aspect) or involved multiple aspects.

3.1 Heuristic Cascade for Positive Examples

To identify positive examples (i.e., revision pairs where significant changes occurred), we apply the following heuristic cascade:

- H_0 (Initial Filter): We verify that $A(q,y) \neq A(q,y+1)$ for a given question q about entity E. This ensures that the value in a Wikipedia list has demonstrably changed between years because the answer to the question has changed in between y and $y+1$.
- H_1 (Structural Similarity): Given the set of *changed* answers, we then filter out those revisions that are too dissimilar. This is to capture cases where the page was empty or merely a stub in year y before becoming a proper article in year $y+1$. We compute the similarity between revisions r_i^y and r_i^{y+1}, requiring it to be high (greater than θ) but not equal to 1. Formally:

$$\text{Sim}(r_i^y, r_i^{y+1}) > \theta \text{ and } \text{Sim}(r_i^y, r_i^{y+1}) \neq 1 \quad (1)$$

  This similarity is computed using `sentence-transformer`[4] based embeddings for the first 512 BERT [9] tokens of each revision. This phase ensures that both r_i^y and r_i^{y+1} are similar enough to guarantee a gradual change (but change nonetheless) in that entity's timeline.
- H_2 (Content Verification): Using LLaMA-3 [13], we verify that the answers from TAQA are contained in their respective revisions with respect to the original question. Formally, for each pair $(r_y, A(q,y))$ and $(r_{y+1}, A(q,y+1))$, we prompt LLaMa to determine **Cont.** as follows:

$$\textbf{Cont.}(r_y, A(q,y) \mid q) = 1 \;\&\&\; \textbf{Cont.}(r_{y+1}, A(q,y{+}1) \mid q) = 1 \quad (2)$$

  This ensures that the core information we are tracking is present in both texts and that the answers correspond to the intended question rather than being incidental mentions. In other words, we ensure that these Wikipedia revisions *are talking* about the events in the original tables, as not all table-worthy information of an entity is present in their corresponding wikipage.
- H_3 (Information Novelty): We verify that the cosine similarity between revision r_y and the concatenation of question q with answer $A(q,y+1)$ is not more than λ:

$$\text{Sim}(r_y, q \oplus A(q,y+1)) \leq \lambda \quad (3)$$

  This ensures there is no evidence in the first revision of the updated information that appears in the second revision, to ensure that when the content related to $A(q,y+1))$ appears in a wikipage's timeline, this is indeed a novel appearance.
- H_4 (Expert Confirmation): As a final verification step, we use a GPT-based validation to confirm that a meaningful change exists between the revisions. This step acts as a quality control measure, filtering out pairs that might have passed earlier heuristics but do not represent substantive changes.

3.2 Distinguishing Atomic vs. Multiple Changes

For revision pairs that have been confirmed to contain changes, we further classify whether these changes are atomic (a single change) or multiple (several changes) with the following H_5 heuristic:

[4] https://sbert.net/: `sentence-transformers/all-MiniLM-L6-v2`.

- We sentence-tokenize both r_y and r_{y+1}, obtaining sets of sentences S_y and S_{y+1}.
- We define heuristic $H_5 : R(E,T) \rightarrow \{\text{a}, \text{m}\}$ as a function that maps a revision pair to a label indicating the type of change:

$$H_5(r_i^y, r_i^{y+1}) = \begin{cases} \text{a} & \text{if } |S_{y+1}| - |S_y| \leq 3 \\ \text{m} & \text{if } |S_{y+1}| - |S_y| > 3 \end{cases} \quad (4)$$

We reach the threshold of 3 for dividing atomic and multiple change by selecting different values, specifically 1, 3 and 5, and manually validating a sample of 100 instances for each. We found that 3 gave the best trade-off between coverage and precision, with 65% of the sample signaling atomic change.

3.3 Identifying Negative Examples

With a set of "positive" instances (wikipage revision pairs at two timestamps with demonstrable change(s) occurring between them), we are now interested in sourcing negative examples. A dataset of this nature (and derived classifiers) has applications in knowledge base maintenance, to detect whether information in a KB needs updating based on evidence from its Wikipedia counterpart; news verification, e.g., for determining if new information about an entity represents a genuine change or just noise; temporal question answering, specifically for supporting systems that must reason about changing facts; or content summarization, for identifying key inflection points in an entity's history.

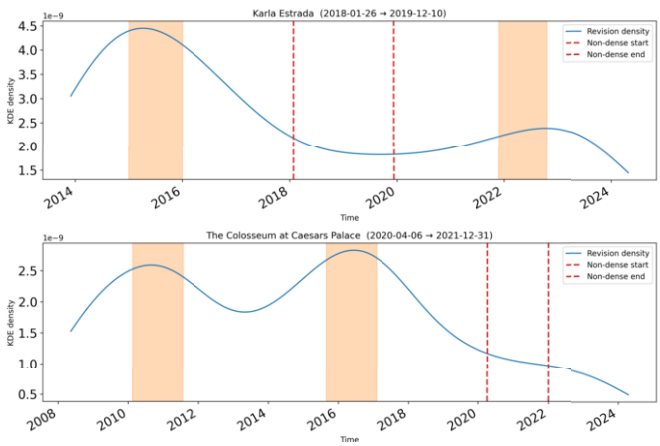

Figure 1: Wikipedia revision history plotted in a timeline, with stable (not captured by KDE) regions shown within two red lines and dense (highly active) regions in yellow shadow. Top: Karla Estrada. Bottom: The Colessium at Ceaser's Palace.

However, obtaining valuable (hard to tell apart) negative examples is not straightforward. It is not sufficient to randomly sample wikipages that are very similar, as this would almost certainly lead in capturing two almost identical snapshots. Therefore, it is important that this phase samples these negatives from period of "low activity", thereby avoiding time periods where an entity's wikipage has undergone significant edits (e.g., a politician's page during a political campaign, where information about rallies, public appearances or electoral promises is constantly added, modified and moderated/removed). Within these downtime periods, we are interested in satisfying the constraint: similar revisions which talk about the table-derived information (in other words, questions and answers from the temporal QA dataset are mentioned). We implement this approach as follows:

1. For each timeline, we identify "moments of change" using a bayesian changepoint detection algorithm, which has been used in the past for finding significant shifts in timelines, e.g., in the context of social media activity [1]. Then, we apply kernel density estimation (KDE) to identify dense regions around change points, which denote active elongated periods around sudden spikes in a wikipage's revision history. We then proceed as follows:
2. We select revision pairs (r_y, r_{y+1}) where both years fall outside these dense regions, ensuring we focus on stable periods. Figure 1 shows two representative examples of revision histories and illustrate *where in those timelines* we sample negatives from. Specifically, we select the regions with stable revision activity and take the revision pairs that fall within that region as our timestamps.
3. We then apply a flipped-sign version of H_0 and H_1. We do not look into exact matching between the texts to identify the information change, but rather try to ensure answer stability and structural similarity via thresholding.

Finally, the final dataset $R(E,T)_l$ for a given entity E and its associated timeline(s) T comprises:

$$R(E,T)_{\text{n}} = \{(r_i^y, r_i^{y+1}) \in R(E,T) \mid H_0^n \wedge H_1^n \wedge H_2^n\} \quad (5)$$

$$R(E,T)_{\text{a}} = \{(r_i^y, r_i^{y+1}) \in R(E,T) \mid \bigwedge_{i=0}^{4} H_i \wedge H_5(r_i^y, r_i^{y+1}) = \text{a}\} \quad (6)$$

$$R(E,T)_{\text{m}} = \{(r_i^y, r_i^{y+1}) \in R(E,T) \mid \bigwedge_{i=0}^{4} H_i \wedge H_5(r_i^y, r_i^{y+1}) = \text{m}\} \quad (7)$$

Thus, for a given entity E and associated timelines T, we have $R(E,T)_l = R(E,T)_{\text{n}} \cup R(E,T)_{\text{a}} \cup R(E,T)_{\text{m}}$, where each pair is linked to its triggering question-answer pairs from the temporal QA dataset. Tables 1 and 2 contain examples of the TACTICAL-derived dataset and summary statistics of the binary classification datasets used in Section 4, respectively.

Data Statistics	Change/No-change			Atomic/Multiple		
	ch.	n.	Total	a.	m.	Total
# instances (r_i^y, r_i^{y+1})	2,327	8,278	10,605	773	1,554	2,327
# of unique titles	627	1,732	2,359	372	519	891
Instances to title ratio	3.7	4.8	-	2.1	2.9	-

Table 2: Statistics of our TACTICAL-derived dataset, showing figures for instances (revision pairs) for change (**ch.**), no change (**n.**), atomic change (**a.**) and multiple change (**m.**).

Label	Heuristic	Accuracy	Instances
Change	H2	0.72	9,859
	H3	0.89	**2,327**
No-change	H1	0.87	16,351
	H2	0.99	**8,278**
Atomic-change	H5 (a)	0.93	773
Multiple-change	H5 (m)	0.95	1,554

Table 3: Human validation results and number of instances filtered for the data collection heuristics.

In Table 3, we summarize the human validation we conducted for different stages of these heuristics. All of them are the result of manually validating the outcome of each one on a random sample of 100

instances, making the manual validation effort comprise 600 distinct revision pairs as deemed relevant for the corresponding goal (e.g., capture change, or capture multiple-aspect change). As we can see, the majority of them are extremely effective, with an accuracy ranging from 0.72 in the worst case (although in practice this lower-than-the-rest accuracy is less impactful due to instances steeming from H2 undergoing further validation in subsequent heuristics), to .99 in the best case scenario[5].

4 Experiment 1: Change Detection and Typification

Our first goal, with a TACTICAL-derived dataset at hand, is to analyze its usefulness to train text classifiers on the tasks of distinguishing meaningful vs. superficial, and atomic vs. multiple changes. As we have seen, such classifiers could impact multiple knowledge management applications, in addition to the time-steering focus we pursue in this paper. To this end, we first take the full datasets (cf. Table 2) and produce four train/validation/test splits. These are *Random* (randomly sampling revision pairs into one split); *No-overlap* (ensuring revision pairs belonging to one entity are all placed in one split, thereby avoiding contamination); and *TFwd* and *TRvsd*, where the cutoff is longitudinal, forward and reversed (see Figure 2 for the data distribution in these two temporal splits from Change/No-change).

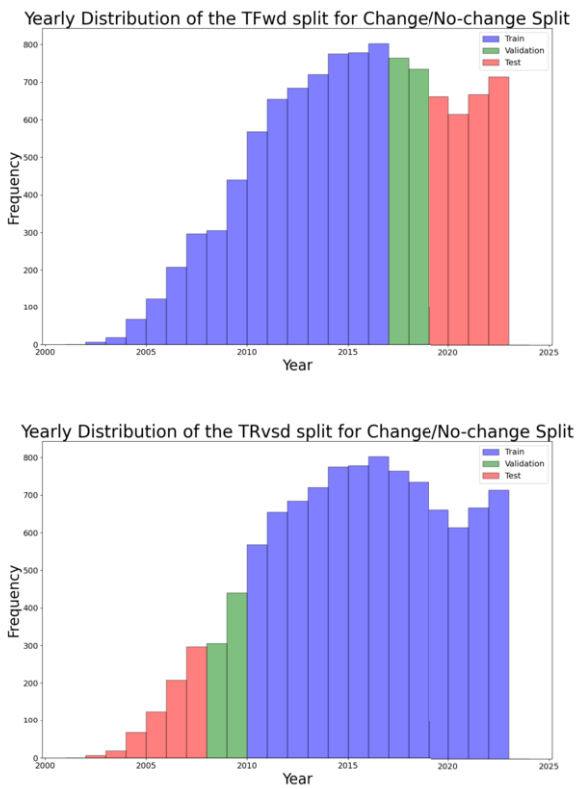

Figure 2: Yearly distribution comparison for Change vs. No change. Top: Forward distribution (TFwd). Bottom: Reversed distribution (TRvsd).

[5] All validation datasets are provided in the supplementary materials. Implementation details can be found in the Github Repository.

With these splits at hand, we fine-tune a number of BERT-based [9] classifiers. We found their performance to be generally competitive (full set of results in Figure 3), with the large models (BERT and RoBERTa) performing highest in change vs. no change, and an unsurprising dip in F1 for the time-capped splits (specifically, TRvsd for change vs. no change; and TFwd for atomic vs. multiple change). Regarding atomic vs. multiple change, we find a surprisingly high performance in the TRvsd split by the ALBERT family of models. Perhaps due to their smaller size, they suffer from less overfitting/bias and are able to learn the dataset patterns better without defaulting to the knowledge learned during pretraining.

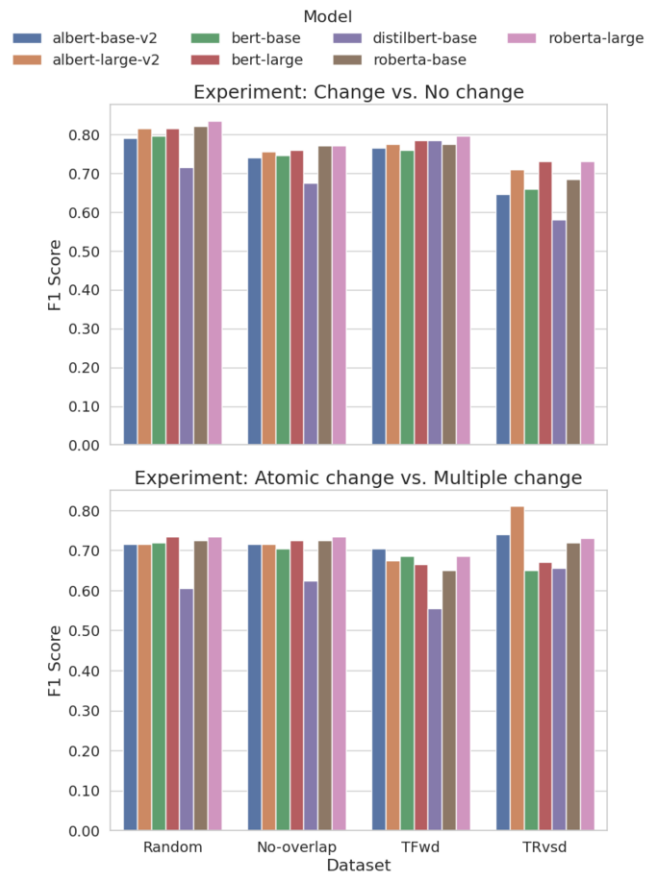

Figure 3: Performance breakdown for different BERT-based models for the different TACTICAL-derived test sets.

5 Expt. 2: LLM Temporal Grounding via TACTICAL Exemplars

It is common practice in the temporal grounding literature to investigate LLMs' behavior when presented with temporal challenges. As we have previously discussed, this has been approached in tasks like temporal question answering (where performance is often measured in F1), as well as running LLMs on out-of-distribution textual samples and measuring their "understanding" of such samples via pseudo-likelihood, perplexity or bits-per-character. In this experiment, we build a novel evaluation dataset for probing LLMs' timeline knowledge of entities and events, which we derive from Wikipedia lists, and show that a frozen pretrained LLM can be better enhanced with TACTICAL-derived exemplars than baseline ICL methods.

5.1 Dataset construction

Our aim in this experiment is to gauge the extent to which LLMs are able to position in a timeline relevant facts surrounding entities and events. Our methodology for probing this aspect involves, first, a Wikipedia list for an entity E (e.g., en.wikipedia.org/wiki/Karla_Estrada#Television), from which we create *true statements* such as "Karla Estrada's role in 2001 was Lilay", and *false statements* such as "Karla Estrada's role in 2021 was Lilay". This is done via GPT prompting, and starting from the entities contained in TAQA. As we can see, false statements are created by swapping the year of a true statement. In order to test models' sensitivity to the plausibility of the false fact, we create three versions of this dataset: plus 1, plus 5 and plus 10, which signal the years the negative examples are off from the true counterpart.

5.2 Prompt Configurations

Following [20], we evaluate frozen LLMs on the binary classification task of determining whether a given statement as derived from the method described in Section 5.1 is true or not. We prompt multiple LLMs and evaluate the effectiveness of different ICL approaches for "setting their clock" and activate their temporal understanding of E via in context learning. We compare multiple prompting methods, which we describe next:

- **Baseline**: A simple prompt where we only provide the description of the task, with no exemplars
- **ICL**: Provide three **true** statements about E in Q&A format as described above
- **TACTICAL**: Provide two past revision pairs for E from TACTICAL, randomly sampled from $R(E,T)_{\text{a}} \cup R(E,T)_{\text{m}}$
- **ICL+TACTICAL**: Combine **ICL** and **TACTICAL** exemplars above
- **ICL+TACTICAL-a**: TACTICAL-derived exemplars are sampled only from $R(E,T)_{\text{a}}$
- **ICL+TACTICAL-m**: TACTICAL-derived exemplars are sampled only from $R(E,T)_{\text{m}}$

We tested four frozen LLMs under each of these prompt settings, namely Llama-3.2-3B and Llama-3.1-8B[13], Gemma-2-2B [32], and Qwen-2.5-7B [35].

5.3 Classification Results

Our results, summarized in terms of F1 in Figure 4, clearly indicate that the knowledge in TACTICAL is more effective for activating temporal knowledge in LLMs than ICL samples. We find that in the hardest setup (+1), Gemma benefits the most from TACTICAL examples, especially when combined with ICL samples. We also observe a significant improvement in Qwen, with all TACTICAL prompts reaching very similar F1. For easier setups, we find a remarkably high performance when combining ICL examples with TACTICAL examples in Gemma. We also find that passing in the prompt exemplars of multiple changes (revision pairs signaling significant changes in one year) produces the best result in about 60% of the cases, whereas TACTICAL-only exemplars seem not to be as effective, and show performance below baseline (no exemplars) or ICL only setups in 80% of experiments.

The main takeaway is clear: Combining TACTICAL exemplars with semi-structured information in the form of temporally grounded Q&A pairs can help LLMs activate their temporal knowledge, with the advantage of not requiring fine-tuning, which enables using this method even for closed-source models only available via API.

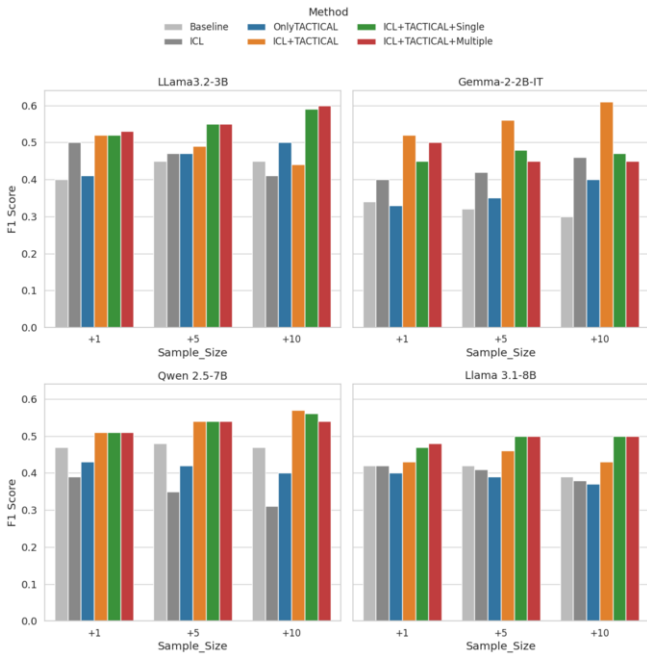

Figure 4: Performance breakdown for different LLMs for the different temporal grounding settings. Best viewed in colour.

5.4 Analyzing year-wise performance

To better understand how TACTICAL-based exemplars enhance temporal abilities in LLMs, we analyzed both accuracy and confidence across different years in our dataset. We divided our data by year and examined Llama 3.2 model predictions under both ICL and ICL+TACTICAL setups. Confidence was measured using the log-probability of the model's first predicted token.

As shown in Figure 5, the model's confidence generally increases with greater temporal distance between true and false instances, with performance peaks visible near the model's training cutoff years. The consistent improvements across different years and splits demonstrate that providing rich temporal context through TACTICAL exemplars effectively enhances the model's ability to make temporally-aware predictions. Moreover, our analysis in Figure 6 reveals significant improvements when using TACTICAL exemplars across years, with the most notable gains occurring when both accuracy and confidence increased simultaneously. A closer look also reveals the following insights: (1) Harder setups cause higher volatility, as we can see from the differences between the easiest +1 split (mostly stable accuracy in the 0.3-0.5 range), but as the temporal gap increases (especially in +10), there is significantly more volatility in model performance, particularly in earlier years (2000-2010); (2) all three splits show a generally improving trend from approximately 2010 onwards. In fact, 2010 seems to act as a "transition year", where in all cases we see a shift from high volatility to stable performance; and (3) high confidence predictions (green points) almost exclusively appear in recent years (2023+) and show the highest accuracy.

5.5 Relationship with popularity

To investigate whether entity popularity influences LLMs' temporal understanding capabilities, we analyzed the correlation between

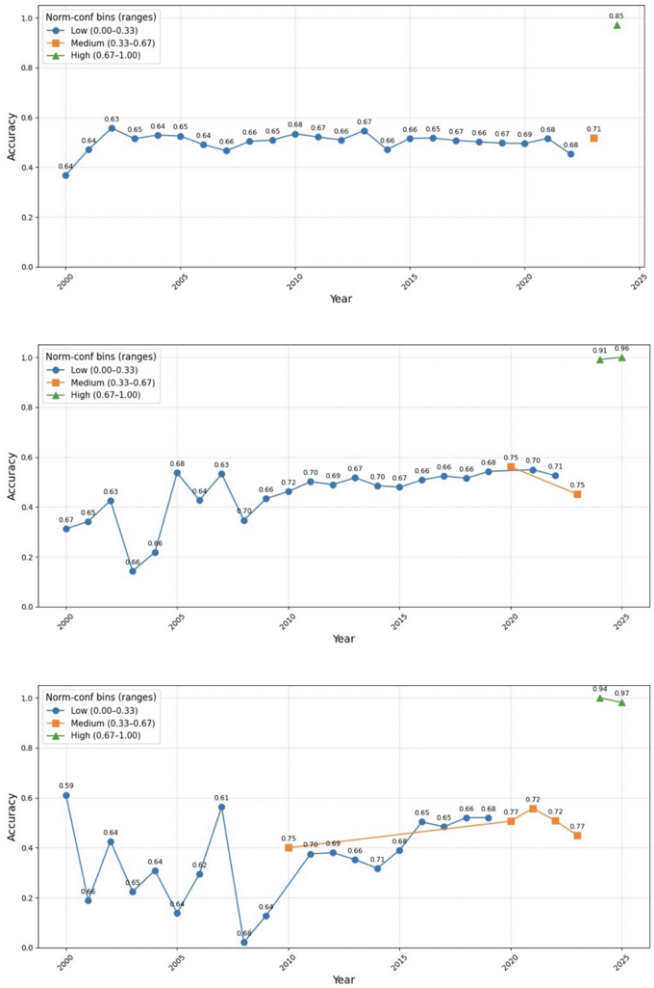

Figure 5: Comparison of accuracy and confidence across years for all three time-splits: +1 (top), +5 (middle) and +10 (bottom). The numbers above each point represent the Accuracy (in the Y-axis) of each data point for the Year (x-axis).

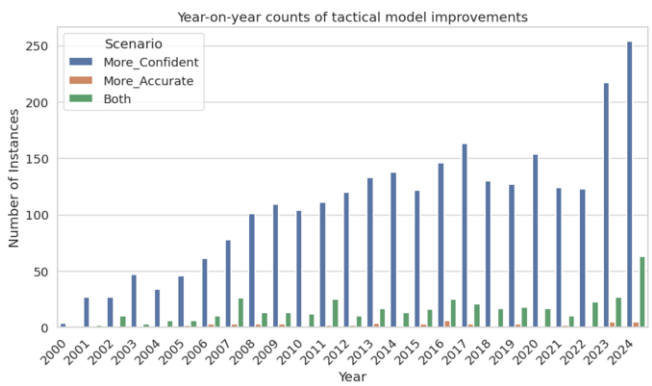

Figure 6: Year-on-year performance (accuracy, model's confidence and both).

Wikipedia pageview statistics and model performance. We collected pageview data for all 100 entity titles in our dataset, spanning from January 1, 2001, to April 30, 2025. For each entity, we calculated the per-title macro F1 scores across three prompting configurations (Baseline, ICL, and ICL+TACTICAL) in the most challenging +1 setting. Table 4 presents performance metrics for the five most and least popular entities based on pageview count, and reveal important patterns. First, both high and low popularity entities show substantial improvements when moving from Baseline to ICL+TACTICAL prompting. High-popularity entities like Kaley Cuoco (38.6M pageviews) show a 14-point accuracy improvement from Baseline (0.38) to ICL+TACTICAL (0.52). Similarly, less popular entities such as Wolfdogs Nagoya (52.9K pageviews) demonstrate even more dramatic improvements, with accuracy increasing by 29 points from Baseline (0.33) to ICL+TACTICAL (0.62). Notably, the most significant relative improvements occur for low-pageview entities, suggesting that TACTICAL-derived exemplars particularly benefit LLMs' temporal understanding of less popular entities. This finding indicates that our approach can effectively address knowledge gaps in the long tail of entity recognition, where information might be less reinforced during pretraining due to lower representation in the corpus. We acknowledge however that pageviews is only a superficial metric for popularity, and leave for future work an analysis of temporal grounding vs, e.g., incoming links[6].

Titles	Baseline	ICL	ICL+T	Pageviews
Kaley Cuoco	0.38	0.51	0.52	38,583,136
Katey Sagal	0.39	0.50	0.51	18,622,787
Chrishell Stause	0.32	0.55	0.54	13,006,802
Gillian Jacobs	0.37	0.54	0.52	11,917,553
Lily Rabe	0.34	0.42	0.47	10,104,843
Titles	**Baseline**	**ICL**	**ICL+T.**	**Pageviews**
Wolfdogs Nagoya	0.33	0.48	0.62	52,927
Harvard Model Congress	0.39	0.47	0.50	51,973
Sean Shields	0.38	0.43	0.53	47,813
Davor Blažević	0.45	0.51	0.51	30,360
Japan Breeding Farms'-Cup Classic	0.33	0.35	0.41	2,261

Table 4: Model performance (Accuracy) on top and bottom 5 titles for different prompting settings (T: TACTICAL).

6 Conclusion and Future Work

In this work, we propose TACTICAL, a heuristic-based framework for the extraction of entity timelines by leveraging Wikipedia's revision histories. We systematically extract a dataset using TACTICAL and conduct text classification experiments to evaluate the model's capability to detect meaningful changes. We also conduct an in-context learning experiment on a set of frozen LLMs to reveal the benefit of using the examples from TACTICAL in improving their temporal fact recall ability. For the future, we would like to homogenize our heuristics to be fully LLM-driven and use LLM-as-a-judge along the way. Another interesting dimension to explore through this work is Knowledge Editing. Despite the coarse-grained year-level granularity in the dataset, we want to develop it further and create a fully fledged temporal KE benchmark made of free-text examples, a largely unexplored area. Developing such resources would provide insights into how factual knowledge evolves and how LLMs can track or prioritize it.

[6] Utilizing projects such as e.g., *qrank*: https://github.com/brawer/wikidata-qrank.

References

[1] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[2] O. Agarwal and A. Nenkova. Temporal effects on pre-trained models for language processing tasks. *Transactions of the Association for Computational Linguistics*, 10:904–921, 2022.

[3] A. Akyürek, E. Pan, G. Kuwanto, and D. Wijaya. Dune: Dataset for unified editing. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 1847–1861, 2023.

[4] H. Borkakoty and L. Espinosa-Anke. Chew: A dataset of changing events in wikipedia. *arXiv preprint arXiv:2406.19116*, 2024.

[5] C. ChenghaoZhu, N. Chen, Y. Gao, Y. Zhang, P. Tiwari, and B. Wang. Is your llm outdated? a deep look at temporal generalization. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 7433–7457, 2025.

[6] R. Cohen, E. Biran, O. Yoran, A. Globerson, and M. Geva. Evaluating the ripple effects of knowledge editing in language models. *Transactions of the Association for Computational Linguistics*, 12:283–298, 2024.

[7] N. De Cao, W. Aziz, and I. Titov. Editing factual knowledge in language models. *arXiv preprint arXiv:2104.08164*, 2021.

[8] M. Del Tredici, R. Fernández, and G. Boleda. Short-term meaning shift: A distributional exploration. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2069–2075, Minneapolis, Minnesota, 2019. Association for Computational Linguistics.

[9] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186, 2019.

[10] B. Dhingra, J. R. Cole, J. M. Eisenschlos, D. Gillick, J. Eisenstein, and W. W. Cohen. Time-aware language models as temporal knowledge bases. *Transactions of the Association for Computational Linguistics*, 10:257–273, 2022.

[11] Q. Dong, D. Dai, Y. Song, J. Xu, Z. Sui, and L. Li. Calibrating factual knowledge in pretrained language models. *arXiv preprint arXiv:2210.12883*, 2022.

[12] P. Fernandes, A. Madaan, E. Liu, A. Farinhas, P. H. Martins, A. Bertsch, J. G. de Souza, S. Zhou, T. Wu, G. Neubig, et al. Bridging the gap: A survey on integrating (human) feedback for natural language generation. *Transactions of the Association for Computational Linguistics*, 11:1643–1668, 2023.

[13] A. Grattafiori, A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

[14] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

[15] J. Jang, S. Ye, S. Yang, J. Shin, J. Han, G. Kim, S. J. Choi, and M. Seo. Towards continual knowledge learning of language models. *arXiv preprint arXiv:2110.03215*, 2021.

[16] J. Jang, S. Ye, C. Lee, S. Yang, J. Shin, J. Han, G. Kim, and M. Seo. Temporalwiki: A lifelong benchmark for training and evaluating ever-evolving language models. *arXiv preprint arXiv:2204.14211*, 2022.

[17] J. Kasai, K. Sakaguchi, R. Le Bras, A. Asai, X. Yu, D. Radev, N. A. Smith, Y. Choi, K. Inui, et al. Realtime QA: What's the answer right now? *Advances in neural information processing systems*, 36:49025–49043, 2023.

[18] A. Lazaridou, A. Kuncoro, E. Gribovskaya, D. Agrawal, A. Liska, T. Terzi, M. Gimenez, C. de Masson d'Autume, T. Kocisky, S. Ruder, et al. Mind the gap: Assessing temporal generalization in neural language models. *Advances in Neural Information Processing Systems*, 34:29348–29363, 2021.

[19] O. Levy, M. Seo, E. Choi, and L. Zettlemoyer. Zero-shot relation extraction via reading comprehension. In *21st Conference on Computational Natural Language Learning, CoNLL 2017*, pages 333–342. Association for Computational Linguistics (ACL), 2017.

[20] J. Liu, P. Yu, Y. Zhang, S. Li, Z. Zhang, R. Sarikaya, K. Small, and H. Ji. Evedit: Event-based knowledge editing for deterministic knowledge propagation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 4907–4926, 2024.

[21] D. Loureiro, F. Barbieri, L. Neves, L. Espinosa Anke, and J. Camacho-Collados. Timelms: Diachronic language models from twitter. pages 251–260, 2022.

[22] K. Luu, D. Khashabi, S. Gururangan, K. Mandyam, and N. A. Smith. Time waits for no one! analysis and challenges of temporal misalignment. *arXiv preprint arXiv:2111.07408*, 2021.

[23] A. Madaan, N. Tandon, P. Clark, and Y. Yang. Memory-assisted prompt editing to improve gpt-3 after deployment. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2833–2861, 2022.

[24] K. Meng, D. Bau, A. Andonian, and Y. Belinkov. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

[25] K. Meng, A. Sen Sharma, A. Andonian, Y. Belinkov, and D. Bau. Mass-editing memory in a transformer. *arXiv preprint arXiv:2210.07229*, 2022.

[26] E. Mitchell, C. Lin, A. Bosselut, C. D. Finn, and C. D. Manning. Fast model editing at scale. *arXiv preprint arXiv:2110.11309*, 2021.

[27] A. Rosenfeld and K. Erk. Deep neural models of semantic shift. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 474–484, New Orleans, Louisiana, 2018. Association for Computational Linguistics.

[28] G. D. Rosin and K. Radinsky. Temporal attention for language models. *arXiv preprint arXiv:2202.02093*, 2022.

[29] A. Susuri, M. Hamiti, and A. Dika. Detection of vandalism in Wikipedia using metadata features – implementation in Simple English and Albanian sections. *Advances in Science, Technology and Engineering Systems Journal*, 2(4):1–7, 2017.

[30] T. Szymanski. Temporal word analogies: Identifying lexical replacement with diachronic word embeddings. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 448–453, Vancouver, Canada, 2017. Association for Computational Linguistics.

[31] C. Tan, G. Zhang, and J. Fu. Massive editing for large language models via meta learning. *arXiv preprint arXiv:2311.04661*, 2023.

[32] G. Team, M. Riviere, S. Pathak, P. G. Sessa, C. Hardin, S. Bhupatiraju, L. Hussenot, T. Mesnard, B. Shahriari, A. Ramé, et al. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*, 2024.

[33] T. Tran, A. Ceroni, M. Georgescu, K. Djafari Naini, and M. Fisichella. Wikipevent: Leveraging wikipedia edit history for event detection. In *International Conference on Web Information Systems Engineering*, pages 90–108. Springer, 2014.

[34] T. Vu, M. Iyyer, X. Wang, N. Constant, J. Wei, J. Wei, C. Tar, Y.-H. Sung, D. Zhou, Q. Le, et al. Freshllms: Refreshing large language models with search engine augmentation. *arXiv preprint arXiv:2310.03214*, 2023.

[35] A. Yang, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Li, D. Liu, F. Huang, H. Wei, et al. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

[36] P. Yu and H. Ji. Information association for language model updating by mitigating lm-logical discrepancy. In *Proceedings of the 28th Conference on Computational Natural Language Learning*, pages 117–129, 2024.

[37] H. Zeng, M. A. Alhossaini, L. Ding, R. Fikes, and D. L. McGuinness. Computing trust from revision history. In *International Conference on Privacy, Security and Trust*, 2006.

[38] M. Zhang and E. Choi. Situatedqa: Incorporating extra-linguistic contexts into qa. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7371–7387, 2021.

[39] B. Zhao, Z. Brumbaugh, Y. Wang, H. Hajishirzi, and N. A. Smith. Set the clock: Temporal alignment of pretrained language models. *arXiv preprint arXiv:2402.16797*, 2024.

[40] Z. Zhong, Z. Wu, C. D. Manning, C. Potts, and D. Chen. Mquake: Assessing knowledge editing in language models via multi-hop questions. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 15686–15702, 2023.

[41] C. Zhu, A. S. Rawat, M. Zaheer, S. Bhojanapalli, D. Li, F. Yu, and S. Kumar. Modifying memories in transformer models. *arXiv preprint arXiv:2012.00363*, 2020.

[42] K. Zielinski, R. Nielek, A. Wierzbicki, and A. Jatowt. Computing controversy: Formal model and algorithms for detecting controversy on Wikipedia and in search queries. *Information Processing & Management*, 54(1):14–36, 2018.