A Fairer Evaluation for the Time Awareness in Instruction-Tuned LLMs

Ernie Chu
schu23@jhu.edu
Department of Computer Science
Johns Hopkins University
Baltimore, MD, USA

Junhyeok Lee
jlee843@jhu.edu
Department of Electrical and Computer Engineering
Johns Hopkins University
Baltimore, MD, USA

Dengjia Zhang
dzhang98@jhu.edu
Department of Computer Science
Johns Hopkins University
Baltimore, MD, USA

Abstract

When asked "Who is the President?", a model's answer must depend on time. Yet most evaluations ignore this. Temporal awareness is a critical and under-evaluated capability of large language models (LLMs), especially for real-world applications where factual accuracy depends on the referenced date. Existing benchmarks such as TimeShift assess temporal reasoning by comparing log probabilities of dated statements, favoring base models tuned for next-token prediction. In this work, we introduce a question-answering-based evaluation framework designed to fairly assess temporal reasoning in instruction-tuned LLMs. Our approach builds on the TimeAware dataset and includes two tasks: (1) precise date prediction and (2) time-conditioned factual question answering. We evaluate a range of open-source LLMs, including both base and instruction-tuned variants, under zero-shot and in-context settings, using strong LLMs as automated judges. Results show that instruction-tuned models lag behind base models in date prediction but perform competitively in question-answering, especially with few-shot prompting. These findings highlight a trade-off introduced by instruction tuning and motivate the need for evaluation protocols and training methods that preserve temporal knowledge in instruction-optimized models. Our code is available at `https://github.com/ernestchu/ssm-project`.

1 Introduction

Large language models (LLMs) have achieved impressive performance across a wide range of natural language tasks, including factual retrieval, reasoning, and interactive dialogue [1, 2, 3, 4]. However, one critical dimension often overlooked in mainstream evaluations is *temporal awareness*, the ability to answer questions whose correct response depends on the referenced date or time. In practical applications, such as news summarization or digital assistants, temporal reasoning is essential. For example, the correct answer to "Who is the US President?" depends entirely on the time at which the question is asked.

To address this gap, prior work by Herel et al. [5] introduced the *TimeAware* dataset and *TimeShift* framework, which evaluate temporal knowledge via log-probability scoring of dated declarative statements. While informative, this evaluation setup implicitly favors base (pretrained) models that are optimized for next-token prediction. Instruction-tuned models (now widely deployed due to their

Final Projects for CS 601.471/671 NLP: Self-supervised Models

superior performance in conversational and question-answering settings) are often disadvantaged by this token-level scoring approach.

In this paper, we present a more user-centric and instruction-aligned framework for evaluating temporal reasoning. Building on the TimeAware dataset, we design two question-answering tasks that reflect how users typically engage with LLMs. The first task asks models to predict the exact date of an event based on its description. The second task simulates real-world use cases by generating natural, time-conditioned factual questions and evaluating the models' answers for correctness.

Our evaluation spans a diverse set of open-source models from the Llama, Gemma, and Qwen families, with both base and instruction-tuned variants assessed in zero-shot and in-context learning (ICL) settings. To ensure fair evaluation, we adopt a cross-model setup where models are judged by other strong LLMs acting as automatic evaluators. We report performance on both precise date prediction and semantic correctness in answering time-aware questions.

Our findings reveal a nuanced trade-off: instruction-tuned models, while well-suited for question answering (QA) formats, still underperform base models in fine-grained date prediction tasks. However, they perform competitively, or even better, in time-conditioned QA, especially when given few-shot examples. These results highlight the need for post-training methods that preserve temporal knowledge while enhancing instruction-following capabilities.

By aligning evaluation more closely with real-world usage, our work contributes a fairer and more practical benchmark for temporal reasoning in LLMs, and opens new directions for designing time-aware and instruction-robust models.

2 Related Work

Several recent efforts have targeted the evaluation of temporal reasoning in large language models (LLMs). Notably, Herel et al. [5] introduced the *TimeAware* dataset and the *TimeShift* framework, focusing specifically on models' abilities to recall and contextualize facts relative to explicit temporal prefixes. Their approach measures model performance based on log probabilities assigned to declarative statements anchored in specific historical contexts, providing valuable insights into how models internalize and utilize temporal knowledge. However, their evaluation primarily benefits base models optimized for token prediction, potentially disadvantaging instruction-tuned models more suited for interactive, instruction-based contexts.

Other prominent works such as *TempReason* [6] and *TRAM* [7] benchmarks have focused broadly on temporal understanding, emphasizing event order, frequency, and duration, rather than precise factual recall at fine-grained temporal granularity. Similarly, *TempLAMA* [8] evaluates models on year-level precision, yet lacks the detailed month or day-level granularity critical for many real-world tasks.

Additional research has explored structural adaptations to model architectures, such as modifications to the self-attention mechanism to explicitly encode temporal information [9]. While these structural enhancements improve performance on tasks like semantic change detection and diachronic embeddings [10, 11], their effectiveness in precisely recalling temporally-bound factual information remains under-examined.

The *Test of Time* benchmark [12] provides another relevant point of comparison by assessing temporal relationships between events. However, it similarly lacks the fine-grained factual recall necessary for evaluating detailed temporal knowledge.

Our work aims to extend and improve upon these previous efforts by introducing a complementary, question-answering-based evaluation method that specifically addresses the limitations observed in instruction-tuned models. By reframing temporal evaluation in a QA format, we seek to achieve a more balanced and realistic assessment across various model architectures, offering deeper insights into the temporal reasoning capabilities essential for dynamic, real-world knowledge applications.

3 Method

To address the potential biases inherent in log-probability-based evaluations of temporal awareness, particularly for instruction-tuned LLMs, we propose a refined method centered on question-answering (QA) paradigms. This approach aims to more accurately reflect real-world user interactions and

Table 1: Examples from the TimeAware dataset [5] showcasing event descriptions with timestamps.

Event	Date
China's government appoints Li Qiang, a close ally of President Xi Jinping ...	2023/03/11
Four people, including the perpetrator, are killed in a vehicle attack in Rochester ...	2024/01/01

provide a fairer assessment of LLMs' ability to recall and utilize time-sensitive factual information. Our method leverages the TimeAware dataset [5], which comprises over 8,000 event descriptions paired with precise timestamps, as illustrated in Table 1.

Our evaluation framework consists of two distinct tasks, each designed to probe different aspects of temporal awareness in LLMs. We ask LLMs to respond with date directly in Section 3.1, and synthesize time-dependent factual questions from event descriptions in Section 3.2, to test if LLMs know what happened on a specific date.

3.1 Task 1: Precise Date Prediction.

This task evaluates the LLM's ability to directly predict the date of an event given its description. We formulate the input as a prompt asking the LLM to provide the date in YYYY/MM/DD format. The specific procedure is laid out as follows:

1. **Prompt Construction:** For each event in the TimeAware dataset, we construct a prompt of the form: *"Answer in YYYY/MM/DD, on what date was the news about the following event published? {Event Description}. Answer: "*.
2. **Output Processing:** For open-source models, we analyze the output probability distribution to ensure the generated date adheres to the specified format. For closed-source APIs, we enforce the format through explicit prompting and discard any outputs that do not comply.
3. **Evaluation Metric:** We assess the accuracy of the predicted date by comparing it to the ground truth date from the TimeAware dataset. A prediction is considered correct if it exactly matches the ground truth.

This task offers a direct measure of the LLM's recall of event timestamps, providing insights into its temporal knowledge retention.

3.2 Task 2: Time-Conditioned Question Answering

This task simulates real-world user interactions by posing factual questions about events within specific temporal contexts. It aims to evaluate the LLM's ability to provide accurate and contextually relevant answers. The detailed procedure is described as follows:

1. **Question Generation:** We employ GPT-4o to generate a factual question based on each event description in the TimeAware dataset. This ensures that the questions are natural and relevant to the event.
2. **Prompt Preparation:** We introduce temporal context by specifying a particular date or time frame in the prompt. For instance, we might ask, "What happened on {Date}?".
3. **Answer Generation:** We test LLMs to generate answers to the posed questions, considering the provided timeframe.
4. **Answer Evaluation:** We utilize open-source models as LLM judges to automatically assess whether the generated answers echo the GT event description from the TimeAware dataset. This evaluation considers the semantic meaning and factual accuracy of the response.

A concrete example to help understanding:

- **Event (Date):** Croatia adopts the euro and joins the Schengen Area. (2023/01/01)
- **Possible Question Generated by GPT-4o:** What major economic and travel-related changes did Croatia implement on January 1, 2023?

- **Possible Answer Generated by LLMs:** On January 1, 2023, Croatia adopted the euro as its official currency and joined the Schengen Area.

This task is specifically designed for instruction-tuned models, as it aligns with their training objectives of answering questions and following instructions. The use of LLMs as evaluators mitigates the subjectivity inherent in open-ended question answering, ensuring a more objective and consistent assessment.

4 Experiment

We run our proposed evaluation on a range of open-sourced models from leading LLMs vendors, including Llama 3 [13], Gemma 2 [14], Gemma 3 [15], Qwen 2.5 [16], and Qwen 3 [1] for both of their pretrained base models (if available) and the instruction-tuned models.

4.1 Task 1: Precise Date Prediction

We curated a list of LLMs with manageable sizes, including both of their base and instruction-tuned models:

- Standard models:
  - Tiny models: Qwen-2.5-3B, Llama3.2-3B and Gemma-2-2B
  - Small models: Qwen-2.5-7B, Llama3.2-8B and Gemma-2-9B
- Latest models:
  - Tiny models: Qwen3-4B and Gemma3-4B
  - Small models: Qwen3-14B and Gemma3-12B

Each model is prompted following the procedure described in Section 3.1. Excluding the manually inserted slash in the *YYYY/MM/DD* format, we sequentially perform 10 next-token predictions to complete each date prediction. For each token prediction, we extract the logits corresponding to digits 0 through 9 and select the digit with the highest logit (argmax) as the predicted token. Notably, no date validation is performed during prediction; therefore, a poorly performing LLM may generate invalid dates such as 2/30 or 13/01. In addition to evaluating models under zero-shot settings, we also assess the impact of in-context learning [1] on performance. Specifically, five random examples from the dataset are held out as demonstrations and prepended to the prompt for each test query.

Metric We provide multiple accuracy measures for holistic evaluation. Specifically, "Y" denotes the percentage of examples where the predicted year matches the ground truth. "YM" refers to examples where both the predicted year and month are correct. "YMD" requires an exact match on year, month, and day. To account for minor errors in day prediction, we also report "YMD $\pm n$D", which measures the accuracy when the predicted date falls within n days of the ground truth date, while remaining within the same calendar month. These complementary metrics allow a more nuanced assessment of the model's temporal reasoning abilities, distinguishing between coarse-grained and fine-grained prediction capabilities.

Results Table 2 summarizes the model performances on precise date prediction, complemented with the visual comparison in Figure 1. Overall, Gemma model family consistently achieves the highest accuracy across all metrics. In the zero-shot setting, Gemma3-12B obtains 67.76% year accuracy and 34.67% exact YMD accuracy, with notable improvements under in-context learning, reaching 75.33% and 40.84% respectively. Qwen model family also performs competitively, though slightly behind Gemma, with moderate gains from in-context learning. In general, the instruction-tuned variants initially underperform the base model in zero-shot but benefit more from few-shot demonstrations in some cases. By contrast, Llama model family shows significantly weaker performance, with minimal improvement even when provided examples, suggesting difficulty in temporal reasoning. Across all models, fine-grained day-level prediction remains substantially harder than coarse year-level matching, and the benefits of in-context learning are most evident in stronger base models.

[1] `https://qwenlm.github.io/blog/qwen3/`

Table 2: Accuracy for date prediction. ZS: zero shot. ICL: In Context Learning. Y: correct year. YM: correct year and month. YMD: correct year, month and day. YMD $\pm n$D: YMD with n days tolerance (within the correct month). Instruction-tuned models are postfixed with **it**.

Model (ZS/ICL)	Y	YM	YMD	YMD ±3D	YMD ±5D	YMD ±10D
Qwen2.5-3B	39.36/44.30	11.85/14.02	2.46/3.38	5.47/6.66	6.82/8.31	9.19/10.75
Qwen2.5-3B-it	39.81/41.71	11.17/11.83	2.63/2.95	5.31/5.99	6.62/7.27	8.64/9.43
Llama3.2-3B	0.00/13.27	0.00/0.64	0.00/0.04	0.00/0.13	0.00/0.15	0.00/0.33
Llama3.2-3B-it	0.55/3.58	0.12/0.19	0.01/0.03	0.04/0.03	0.04/0.04	0.07/0.07
Gemma2-2B	47.52/52.56	19.65/20.49	6.09/7.68	10.60/12.30	12.22/13.68	15.57/16.73
Gemma2-2B-it	44.50/49.69	17.19/17.72	4.31/5.15	8.73/9.57	10.16/11.04	13.52/13.81
Qwen2.5-7B	48.98/53.96	22.16/23.81	8.78/9.98	14.42/15.07	15.91/16.72	18.74/19.86
Qwen2.5-7B-it	47.96/49.29	20.88/21.41	7.36/7.62	12.66/12.82	14.41/14.72	17.30/17.84
Llama3.1-8B	2.37/28.54	0.44/2.94	0.01/0.32	0.07/0.77	0.10/1.00	0.15/1.47
Llama3.1-8B-it	8.65/26.36	1.86/2.88	0.09/0.25	0.42/0.65	0.61/0.89	0.86/1.53
Gemma2-9B	66.89/69.18	49.15/51.05	29.39/33.58	41.68/43.14	43.62/45.22	46.21/48.01
Gemma2-9B-it	62.74/64.19	45.04/45.89	25.09/26.56	36.17/37.33	38.48/39.60	41.54/42.67
Qwen3-4B	43.09/44.83	14.36/15.95	4.15/5.44	7.86/8.89	9.23/10.44	11.43/13.05
Qwen3-4B-it	37.07/38.51	11.73/12.20	2.41/2.94	5.38/6.15	6.79/7.43	9.15/9.92
Gemma3-4B	53.53/57.17	24.26/24.53	9.07/10.61	15.08/16.16	16.97/18.03	20.56/21.08
Gemma3-4B-it	46.60/48.41	18.34/18.71	4.81/5.27	9.38/9.76	11.14/11.70	14.12/14.71
Qwen3-14B	53.47/58.89	30.11/32.17	14.88/16.65	22.22/24.26	23.77/26.03	26.88/29.13
Qwen3-14B-it	51.75/55.34	27.82/28.55	12.31/13.19	19.43/20.62	21.11/22.36	24.38/25.41
Gemma3-12B	67.76/75.33	53.29/59.77	34.67/40.84	46.08/52.54	48.12/54.64	50.82/57.29
Gemma3-12B-it	64.74/68.72	45.27/47.62	23.96/24.42	34.39/35.88	36.77/38.59	40.74/42.39

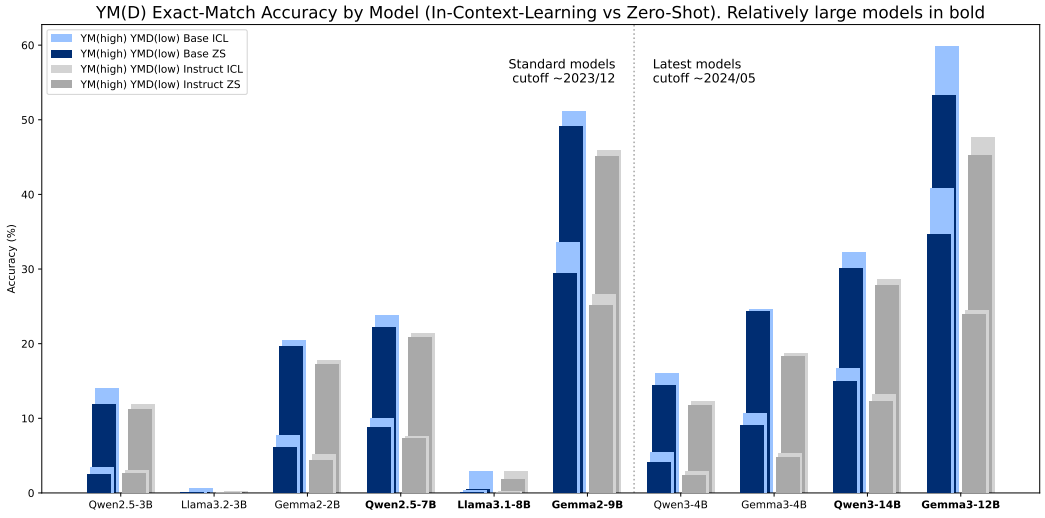

Figure 1: Visual comparison for overall performance. Base models are in blue and Instruction-tuned models are in gray. Models using in-context learning are in lighter shade, and the ones using zero-shot prediction are in darker shade. Since YM scores are always higher then YMD scores by definition, the higher and lower two bars are for YM and YMD scores, respectively.

While we initially expected instruction-tuned models to perform more competitively under the QA-based evaluation framework, our results show that they still lag slightly behind their base counterparts in most cases. This again illustrates from a different perspective that instruction tuning may inadvertently compromise some of the model's pre-existing temporal reasoning abilities.

Original Prompt	At {year}-{month}-{day}, the event {description} happened in {country}, {continent} which is related to {category}. Can you ask a time-aware question that would lead to this event as the answer?
Formatted Prompt	At 2024-1-1, the event "A magnitude 7.6 earthquake strikes Japan's western coast, killing an estimated 120 people and injuring more than 100." happened in Japan, Asia, which is related to Environment & Ecology. Can you ask a time-aware question that would lead to this event as the answer?
GPT-4o-mini	What significant natural disaster occurred in Japan on January 1, 2024, that resulted in numerous casualties?
GPT-4.1-nano	Which environmental disaster occurred in Japan's western coast on January 1, 2024, resulting in approximately 120 fatalities and over 100 injuries?

Table 3: Accuracy of Different Models

Model (ZS/ICL)	Qwen Judge		Gemma Judge	
	Base	Instruct	Base	Instruct
Qwen3-4B	61.56/63.69	62.57/61.91	89.01/93.27	96.51/94.42
Gemma-3-4B	64.03/70.77	62.18/66.50	91.78/98.22	96.46/98.61
Qwen3-14B	76.57/77.42	76.78/80.05	91.30/95.06	86.06/95.13
Gemma-3-12B	80.98/81.66	79.63/78.60	93.60/98.50	95.63/97.35

4.2 Task 2: Time-Conditioned Question Answering

This task simulates real-world user interactions by posing factual questions about events within specific temporal contexts. We first introduce how we synthesize the factual questions (Section 4.2.1), then outline our evaluation pipeline using these questions (Section 4.2.2).

4.2.1 Time-Conditioned Question Generation

Before automatically generating the question, we conducted a brief subjective evaluation comparing GPT-4o-mini and GPT-4.1-nano as Table 4.2.1. We selected GPT-4.1-nano because its outputs consistently contained richer detail and better clarity. Leveraging the OpenAI API with the GPT-4.1-nano model, we extracted from the TimeAware dataset [5] the specific question that would produce each target formulation as its answer. Since TimeAware provides five alternative formulations per entry, we randomly sampled one per row and reformatted it to match the original prompt layout shown in Table 4.2.1.

4.2.2 Time-Conditioned Question-Answering Results

To mitigate potential biases introduced by self-reinforcing model behavior, we adopt a cross-model evaluation protocol. This design ensures that no model is solely evaluated by a judge from the same model family, reducing the risk of stylistic bias in the evaluation process.

Table 3 summarizes the accuracy of different LLMs on the QA-based temporal reasoning task, evaluated using both Qwen3-14B and Gemma3-12B as independent judges. Each LLM's output was assessed for semantic equivalence to the corresponding ground truth event description from the TimeAware dataset. Similar to Task 1, we evaluate both zero-shot (ZS) and in-context learning (ICL) settings. Each question is paired with five random few-shot examples in the ICL condition.

The results reveal several important trends. First, all models benefit from in-context learning, with the strongest improvements observed in the smaller models. For example, Gemma-3-4B (base) improves from 64.03% to 70.77% under Qwen judgment, and from 91.78% to 98.22% under Gemma judgment. This suggests that temporal reasoning capabilities can be significantly boosted by providing demonstrations, especially for instruction-tuned variants.

Second, instruction-tuned models generally match or slightly outperform their base model counterparts in this task, indicating that instruction tuning is well-aligned with the QA-based evaluation

format. For example, Qwen3-14B-it achieves 80.05% under Qwen judgment and 95.13% under Gemma judgment, both slightly higher than its base model in the ICL setting.

Third, the choice of evaluator affects absolute accuracy but preserves overall ranking trends. Judgments from Gemma3-12B are consistently more lenient, yielding higher scores across all models, while Qwen3-14B provides stricter assessments. This highlights the importance of evaluator calibration and supports our use of cross-model validation.

Overall, this task confirms the advantage of instruction-tuned models in natural question answering formats. In-context learning further enhances model performance, particularly for smaller models. However, the observed variability in judgment outcomes underscores the necessity of using multiple or stronger evaluators to ensure robust and reliable assessments.

5 Conclusion

We introduced a question-answering-based framework to evaluate temporal reasoning in large language models (LLMs), focusing on fairness and practical alignment with instruction-tuned models. While prior benchmarks such as TimeShift emphasized log-probability evaluations favoring base models, our framework reframes the task in a natural QA format, better suited for real-world user interactions and instruction-tuned capabilities.

Despite these methodological improvements, our results consistently show that base models outperform instruction-tuned counterparts in precise date prediction tasks, especially under zero-shot settings. This suggests that instruction tuning may inadvertently diminish certain factual recall abilities, including fine-grained temporal precision. Nevertheless, instruction-tuned models show competitive performance in the QA-style evaluations, particularly under in-context learning, where they often match or slightly exceed their base model counterparts.

These findings indicate a nuanced trade-off introduced by instruction tuning: while it improves usability and performance in conversational contexts, it may compromise raw factual precision. Our work highlights the importance of post-training methods that preserve temporal knowledge while enhancing instruction-following abilities.

Overall, this study underscores the need for more balanced and realistic evaluation protocols that reflect how LLMs are actually deployed. It also opens the door for future research into training strategies that preserve time-sensitive knowledge, the development of more robust evaluation metrics, and the use of stronger or ensemble-based LLM judges to ensure reliable automatic assessment.

6 Limitation and Future Work

One limitation of our current study is the reliability of the automatic evaluation. We used open-source models like Qwen3-14B and Gemma3-12B to judge the answers, but these models are not always consistent. In many cases, they marked answers as correct even when the response simply repeated part of the question or used vague language. This led to inflated scores that may not truly reflect the model's understanding. In the future, we plan to use stronger and more trustworthy models, such as proprietary APIs like GPT-4 or Claude, as judges, depending on resource availability. These models can provide more accurate and stricter evaluations.

We also limited our experiments to models with fewer than 14 billion parameters due to computational constraints. This means we did not evaluate many of the strongest available. In future work, we plan to expand our evaluation to include larger models and closed-source APIs to better understand how temporal reasoning scales with model size and architecture.

Finally, our current question generation process relies heavily on GPT-4.1-nano, which may introduce its own biases or limitations in question quality. We aim to explore other approaches, such as using multiple generators or human-written questions, to improve the diversity and realism of the QA tasks.

Overall, there is much room for improvement. More powerful judges, a wider range of models, and better question design will all help build a stronger and more accurate benchmark for testing time awareness in LLMs.

7 Team Work

Ernie Chu conducted the Task 1 experiments, Junhyeok Lee generated factual questions for Task 2 using the OpenAI API, and Dengjia Zhang executed the remaining Task 2 evaluations. All authors contributed to the paper writing and poster design.

References

[1] Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.

[2] OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, et al. Gpt-4 technical report. Technical report, OpenAI, 2024. URL `https://arxiv.org/abs/2303.08774`.

[3] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. LLaMA: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[4] Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. BEIR: A heterogeneous benchmark for zero-shot evaluation of information retrieval models. *arXiv preprint arXiv:2104.08663*, 2021.

[5] David Herel, Vojtech Bartek, Jiri Jirak, and Tomas Mikolov. Time awareness in large language models: Benchmarking fact recall across time. *arXiv preprint arXiv:2409.13338*, 2024. URL `https://arxiv.org/abs/2409.13338`.

[6] Qingyu Tan, Hwee Tou Ng, and Lidong Bing. Towards benchmarking and improving the temporal reasoning capability of large language models. *arXiv preprint arXiv:2306.08952*, 2023.

[7] Yuqing Wang and Yun Zhao. Tram: Benchmarking temporal reasoning for large language models. *arXiv preprint arXiv:2310.00835*, 2024.

[8] Bhuwan Dhingra, Jeremy R. Cole, Julian Martin Eisenschlos, Daniel Gillick, Jacob Eisenstein, and William W. Cohen. Time-aware language models as temporal knowledge bases. *Transactions of the Association for Computational Linguistics*, 10:257–273, 2022. ISSN 2307-387X. doi: 10.1162/tacl_a_00459. URL `http://dx.doi.org/10.1162/tacl_a_00459`.

[9] Guy D. Rosin and Kira Radinsky. Temporal attention for language models. In Marine Carpuat, Marie-Catherine de Marneffe, and Ivan Vladimir Meza Ruiz, editors, *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1498–1508, Seattle, United States, July 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-naacl.112. URL `https://aclanthology.org/2022.findings-naacl.112/`.

[10] Dominik Schlechtweg, Barbara McGillivray, Simon Hengchen, Haim Dubossarsky, and Nina Tahmasebi. SemEval-2020 task 1: Unsupervised lexical semantic change detection. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1–23, December 2020.

[11] William L. Hamilton, Jure Leskovec, and Dan Jurafsky. Diachronic word embeddings reveal statistical laws of semantic change. *arXiv preprint arXiv:1605.09096*, 2018.

[12] Bahare Fatemi, Mehran Kazemi, Anton Tsitsulin, Karishma Malkan, Jinyeong Yim, John Palowitch, Sungyong Seo, Jonathan Halcrow, and Bryan Perozzi. Test of time: A benchmark for evaluating llms on temporal reasoning. *arXiv preprint arXiv:2406.09170*, 2024.

[13] Aaron Grattafiori et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

[14] Gemma Team. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*, 2024.

[15] Gemma Team. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*, 2025.

[16] An Yang et al. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2024.