A Study into Investigating Temporal Robustness of LLMs

Anonymous ACL submission

Abstract

Large Language Models (LLMs) encapsulate a surprising amount of factual world knowledge. However, their performance on *temporal questions* and historical knowledge is limited because they often cannot understand temporal scope and orientation or neglect the temporal aspect altogether. In this study, we aim to measure precisely how robust LLMs are for question answering based on their ability to process temporal information and perform tasks requiring temporal reasoning and temporal factual knowledge. Specifically, we design eight time-sensitive robustness tests to check the sensitivity of six popular LLMs in the zero-shot setting. Overall, we find LLMs lacking temporal robustness, especially in terms of robustness to temporal reformulations and the use of different granularities of temporal references. We show how a selection of these eight tests can be used automatically to judge a model's temporal robustness for user questions on the fly. Finally, we apply the findings of this study to improve the temporal QA performance by up to 98%.

1 Introduction

Despite the strong zero- and few-shot performance of LLMs, it has been recently pointed out that LLMs suffer from a partial or imprecise understanding of the *temporal scope, orientation, and reasoning* expressed in text (Chan et al., 2024; Yuan et al., 2023; Wallat et al., 2024; Jain et al., 2023). The inaccurate understanding of temporal orientation and grounding raises concerns regarding the effectiveness of LLMs over a range of tasks involving temporal reasoning and intents like question-answering and search over historical sources (Wang et al., 2022), QA over legal and personal temporal collections (Qin et al., 2020; Zamani et al., 2017; Gupta et al., 2019), or fact checking (Lee et al., 2020; Nakov et al., 2021). Moreover, questions with temporal aspects are relatively rare in many current question-answering datasets and may thus go undetected in offline evaluations. In this paper, we study the ability of large-language models (Brown et al., 2020) in temporal question-answering tasks given their typically excellent ability of language understanding and reasoning.

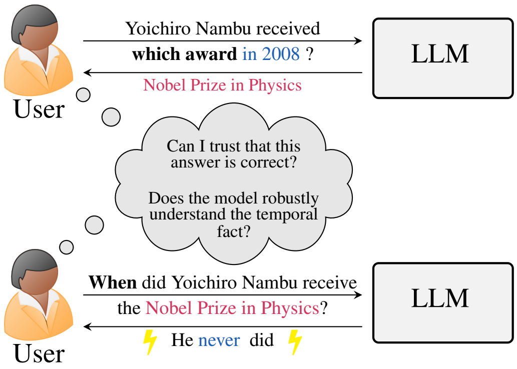
Yoichiro Nambu received
which award in 2008 ?
LLM
Nobel Prize in Physics
User
Can I trust that this answer is correct?
Does the model robustly understand the temporal fact?
When did Yoichiro Nambu receive the Nobel Prize in Physics?
LLM
He never did
User

Figure 1: We investigate the robustness of temporal understanding with a set of tests (here: temporal reversal). By asking the inverse question and looking for consistency between the two answers, we can study if the model understands the temporal-factual information.

Consider the following question: "Who was the prime minister of Pakistan in 1992?". The answer to this question is *Nawaz Sharif*. When asked this question to an LLM (say a Mistral-7B model) by just changing the year *1992* to *1995*, *2010*, or *1970* – we still observe the same response. This simple test indicates a disregard for the time information, possibly due to popularity bias. In this paper, we propose a series of tests that help identify when LLMs can fail due to improper understanding of time and handling of temporal information in question-answering (Figure 1). Unlike earlier literature that focuses on characterizing temporal failures (Wallat et al., 2024), we provide concrete test cases and question reformulations to automatically determine the sensitivity of LLMs to temporal information in questions or lack thereof. This set of tests can be used as LLMs' pre-deployment tests in combination with the regular task performance.

In this paper, we first introduce a range of au-

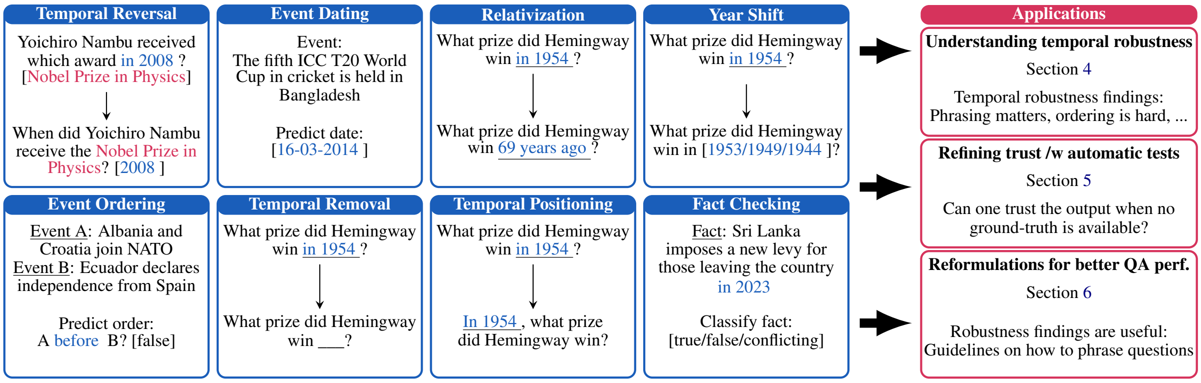
Temporal Reversal
Yoichiro Nambu received which award in 2008 ? [Nobel Prize in Physics]
When did Yoichiro Nambu receive the Nobel Prize in Physics? [2008]
Event Dating
Event: The fifth ICC T20 World Cup in cricket is held in Bangladesh
Predict date: [16-03-2014]
Relativization
What prize did Hemingway win in 1954 ?
What prize did Hemingway win 69 years ago ?
Year Shift
What prize did Hemingway win in 1954 ?
What prize did Hemingway win in [1953/1949/1944]?
Event Ordering
Event A: Albania and Croatia join NATO
Event B: Ecuador declares independence from Spain
Predict order: A before B? [false]
Temporal Removal
What prize did Hemingway win in 1954 ?
What prize did Hemingway win ___?
Temporal Positioning
What prize did Hemingway win in 1954 ?
In 1954 , what prize did Hemingway win?
Fact Checking
Fact: Sri Lanka imposes a new levy for those leaving the country in 2023
Classify fact: [true/false/conflicting]
Applications
Understanding temporal robustness
Section 4
Temporal robustness findings: Phrasing matters, ordering is hard, ...
Refining trust /w automatic tests
Section 5
Can one trust the output when no ground-truth is available?
Reformulations for better QA perf.
Section 6
Robustness findings are useful: Guidelines on how to phrase questions

Figure 2: Overview of the different tests in our temporal robustness test suite. We suggest a suite of several tests that are useful in multiple applications: 1) helping to assess the temporal robustness of LLMs for temporal QA, 2) Calibrating user trust at inference time, and 3) as guidelines on how to reformulate arbitrary (temporal) questions to improve QA performance.

tomatic transformations that manipulate temporal information in questions to estimate the robustness of LLMs towards temporal questions. Our transformations consider multiple and diverse interventions of the time component in questions like *temporal removal, positioning, year-shift*, and *temporal reversal* (cf. Figure 2). Our tests are grounded on well-understood properties and challenges of event-based question answering and temporal information retrieval like temporal ordering and dating (Campos et al., 2014; Kanhabua et al., 2015; Setty et al., 2017). Secondly, we perform extensive benchmarking over three well-known temporal QA datasets (Chen et al., 2021; Wang et al., 2021a, 2022), two additional test sets composed of sentences containing a series of temporal facts and important historical events, and six popular LLMs (Brown et al., 2020). Our results show that although LLMs can answer the questions in their original form, they struggle under certain temporal transformations. Specifically, we see performance drops of 45-71% under the temporal reversal query transformation. Although LLMs can successfully date many events on the year granularity, they find it hard to date them on a day granularity (performance drops by 53-83%). Also, and surprisingly, although LLMs can perform event dating reasonably well, they find it difficult to order events chronologically. Additionally, the findings of this study can be applied and be valuable beyond informativeness: We showcase that a subset of the tests can be used automatically to understand better whether the model's answer is correct without access to the ground-truth answers. Lastly, we show that by applying the findings from this study to new temporal questions, we can use guided reformulations to improve the relative QA performance by up to 98%. Our proposal of benchmarking allows for more precise gauging of how robust LLMs are when it comes to the temporal knowledge and abilities they possess. The findings are applicable in multiple scenarios, and code & data are available[1].

2 Related Work

2.1 Time-aware Pre-trained Language Models

The pretraining approaches used in Language Models (e.g., BERT (Devlin et al., 2019)) do not specifically consider or model temporal information. Several time-focused enhancements and adaptations of language models have been then proposed recently. A naive approach relies on training different versions of a language model on time-segmented portions of data (Qiu and Xu, 2022). This results in multiple language models that require an alignment stage as postprocessing. Other solutions explore dynamic word embeddings (Yao et al., 2018).

More advanced approaches incorporate temporal knowledge during the pretraining stage (Giulianelli et al., 2020; Dhingra et al., 2022; Rosin et al., 2022; Wang et al., 2023). A simple yet effective modification to pre-training is proposed in (Dhingra et al., 2022), where the masked language modeling (MLM) objective is parameterized with timestamp information. Rosin and Radinsky (2022) propose to enhance the self-attention mechanism by integrating timestamp information for updating attention scores. A time-aware prompting strategy for text generation has been proposed by (Cao and Wang, 2022). In a more recent study, Cole et al. (2023)

[1] https://anonymous.4open.science/r/temporalrobustness-B3D3/

Dataset	Example Question	Answer	#Qs	Scope	Description
ArchivalQA	What was Ankara's official aid bill for in 1997?	Cyprus	7,500	1987-2007	detailed quest.
Wikidata	Yoichiro Nambu received which award in 2008?	Nobel Prize	10,000	1907-2018	people
Temporal Claims	China reports military clash in Henan province in 2022	False	4,196	1000-2023	claims
Wikipedia Events	Former Pope Benedict XVI dies at the age of 95.	Dec. 31 2022	23,550	1750-2023	events

Table 1: Overview of the temporal source datasets used to create the robustness tests in this study.

integrate content time into the transformer encoder-decoder architecture (T5 model). They mask time expressions in content and conduct experiments on different temporal tasks. Wang et al. (2023) utilize transformer encoders to leverage both the timestamp and content time (i.e., temporal expressions) in three novel pre-training tasks: document timestamping, temporal expression masking, and temporal information swapping.

2.2 Benchmarking LLMs Robustness

Several studies have examined the temporal reasoning capabilities of LLMs (Wang and Zhao, 2024; Xiong et al., 2024; Chu et al., 2024; Jain et al., 2023; Zhou et al., 2019; Fatemi et al., 2024). Yuan et al. (2024) studied explanatory capabilities of LLMs when forecasting future events, while Chan et al. (2024) analyzed Chat-GPT on inter-sentential relations including temporal and causal relations. Chu et al. (2024) introduced a hierarchical temporal reasoning benchmark called TimeBench and focused on Chain-of-Thought prompting. Wang and Zhao (2024) introduced TRAM, a temporal reasoning benchmark encompassing temporal aspects of events such as order, arithmetic, frequency, and duration. For interested readers, temporal commonsense reasoning datasets and approaches have been overviewed in (Wenzel and Jatowt, 2023).

Temporal factual knowledge has also been the focus of several recent QA datasets created to analyze the performance of LLMs (Chen et al., 2021; Wang et al., 2022; Dhingra et al., 2022; Gruber et al., 2024; Jia et al., 2018, 2024; Mousavi et al., 2024). For example, a diagnostic dataset *TempLAMA* introduced in Dhingra et al. (2022) contains 50k temporally-scoped subject-object relations collected from the snapshot of Wikidata and provided in the cloze-style queries. The authors discuss potential problems related to encoding factual temporal knowledge, such as averaging, forgetting, and poor temporal calibration. More recently, Wallat et al. (2024) study if LLMs can answer temporal questions and reveal that they struggle with simple perturbations in questions like time relativization or time shift. However, the authors do not introduce a complete test suite for temporal robustness as we do (e.g., event ordering, event dating, fact verification, time positioning, temporal reversal), neither propose automatic question transformations nor demonstrate how temporal QA performance can be improved. Bajpai et al. (2024) introduce temporally consistent factuality probing and the corresponding dataset constructed from a knowledge graph for measuring the temporal consistency of objects and their relations. In another work, Beniwal et al. (2024) demonstrate that diverse fine-tuning approaches significantly improve the performance of open-source LLMs, reducing errors caused by knowledge gaps.

Our research emphasizes novel approaches for investigating temporal signals, anchoring knowledge in time, and navigating and orienting over timelines utilizing a range of different datasets and models. We propose a set of automatic transformation steps that, given any temporal QA dataset, allow it to be extended to gauge the temporal robustness of LLMs, and we also demonstrate how our approach can enhance QA performance.

3 Study Details

We use several data sources to test factual knowledge with time-scoped questions for assessing LLMs' robustness in handling temporal references: WikiData (Time-Sensitive QA (Chen et al., 2021)), the historical New York Times news archive (ArchivalQA (Wang et al., 2022)), TemporalQuestions (Wang et al., 2021a), major world events from Wikipedia (event dating/ordering), and fact-checked temporal claims crawled from various websites (Temporal Claims (Venktesh et al., 2024)). An overview is given in Table 1. We elaborate further on the source datasets and model & implementation details in Appendix F.

Time Relativization, Removal, Year Shift, Positioning. We sample 3k QA pairs from ArchivalQA that end with a year reference (e.g., "in 2019?") and modify the references according to the task (as in (Wallat et al., 2024)). For relativization, we convert an absolute year reference to a relative one[2]. For

[2] Using the answer to the question "What year do we have?"

the year shift, we randomly decide whether we add or deduct k years from the question's original year. Lastly, for the positioning test, we move the time reference from the end of the question (e.g., "... in 2019?") to the front of it ("In 2019, ..."). Examples of these transformations and the remaining tests can be seen in Figure 2.

Temporal Reversal. We use WikiData information (similar to Saxena et al. (2021)) and interpret these factual statements as quadruples ($subject$, $relation$, $object$, $time$). In other studies, these quadruples have been used to construct questions such as "Who was the American president in 2012?" Answer: "Obama," asking for the subject or the object of the quadruple. We hypothesize that a thorough, actual understanding of the question's temporal aspect would result in the model being able to answer both the normal (forward) question as well as a reformulation of this question that queries for the time (e.g., "When was Obama president of the USA?" Answer: "2009-2017"). We utilize 10k examples from WikiData for this test since it has quadruples with individual years and the required intervals in which the relation was true. We apply the set of relations used by Saxena et al. (2021) and write templates for the reformulations.

Temporal Fact Checking. We use a dataset of manually verified facts crawled from various verification websites (Venktesh et al., 2024), containing 4,196 temporally scoped claims. Fact verification requires the model to produce a judgment of $true$, $false$, or $contradicting$ for a given claim.

Event Dating/Ordering. Similar to (Wang et al., 2021b), we crawl events from the Wikipedia year pages[3] to acquire fine-grained dates (containing a day, month, and year) and short descriptions of major events between 1750 and 2023. We then filter out events that contain years in the description, as these would be easy to guess. For the event dating test, we ask the model to reproduce the date for a given event in different granularities: year, month, and date. For the event ordering test, we randomly sample events from the same year or for given distances k. We then ask the models to answer which event happened first. The event dating task uses 3k events for each granularity, and the event dating has 3k event pair comparisons.

Evaluation. We utilize a set of model-specific metrics (OpenEval and answer equivalence (Kamalloo et al., 2023; Bulian et al., 2022)) and model-agnostic metrics (i.e., token recall and answer string containment (Adlakha et al., 2024; Liu et al., 2024; Mallen et al., 2023)). OpenEval evaluates the correctness of an answer by querying whether a candidate is a suitable answer given the question and the reference answer[4]. The BEM metric uses a BERT model trained on human-labeled data to predict equivalence between a candidate answer and a reference given a question.

4 Testing Temporal Robustness

In the upcoming sections, we investigate different classes of temporal questions and problems and how our models react to them. For convenience, we show an overview of all models and temporal robustness results in Table 2.

4.1 Time Relativization

The first test that we apply is measuring the effect that switching the time reference from an absolute one (e.g., "2019") to a relative one (e.g., "5 years ago") has on the models' ability to answer temporal factual questions. Relative temporal expressions are a common way to refer to time points, especially when one wants to emphasize the duration of elapsed time. Given that our models can all perform the reasoning needed[5], one would expect the LLMs to be robust to this paraphrase. Thus, an ideal model should perform on par for both absolute and relative questions (results in Table 3).

Interestingly, out of the 39.2% of questions that `Llama 3.1` can answer without paraphrasing, only 22.3% are also answered correctly when using the relative time reference, resulting in a decrease of 43%. We observe similar performance decreases in the other models (21-43%). Specifically, the more capable models seem to be more (but not entirely) robust to using relative references. Given that *all models lack robustness w.r.t. relative time references*, we question how much of the models' performance is due to statistical parroting or a profound and usable understanding of the factual information and the corresponding time component.

4.2 Time Removal

In the time removal test, we study the relevance of the temporal reference on the question-answering

as a reference year.

[3] E.g., https://en.wikipedia.org/wiki/2006

[4] For which we utilize `Flan-T5-XXL`

[5] Which we verify by asking, "What year was 5 years ago?"

Model	**Size**	**↔Relativ.**	**↓Removal**	**Shift**	**↔Reversal**	**↑Facts**	**↔Date**	**↔Order**	**↔Position**
Llama 3.1	8B	↓43%	↓51%	↓26%	↓61%	29.1	↓83%	↑6%	↑34%
Gemma 2	27B	↓32%	↓49%	↓29%	↓62%	39.9	↓71%	↓16%	↓28%
Qwen 2.5	32B	↓42%	↓50%	↓44%	↓71%	74.7	↓64%	↓32%	↑21%
Jamba 1.5	52B	↓21%	↓38%	↓25%	↓55%	65.5	↓71%	↓2%	↑43%
Cmd-R+	104B	↓35%	↓44%	↓30%	↓50%	46.5	↓57%	↓16%	↑21%
GPT 4	unk.	↓32%	↓46%	↓41%	↓45%	33.1	↓53%	↓25%	↑16%

Table 2: Overview of the temporal robustness tests. If there is a clear preference, we denote the tests with ↔ / ↓ / ↑ (in table header) to indicate whether we expect well-performing models to be oblivious/decrease/increase in performance on this task. For example, we expect a robust model to be oblivious to relativization and to stay constant in its performance.

Model		**↔Relativization**		**↓Removal**	
	Abs	Abs ∩ Rel.	Diff.	Abs ∩ Rem.	Diff.
Llama 3.1	39.2	22.3	↓43%	19.1	↓51%
Gemma 2	40.6	27.8	↓32%	20.6	↓49%
Qwen 2.5	32.9	19.2	↓42%	16.4	↓50%
Jamba 1.5	47.0	**37.1**	↓**21%**	**29.2**	↓38%
Cmd-R+	**47.5**	30.9	↓35%	26.7	↓44%
GPT 4	**47.5**	32.2	↓32%	25.9	↓46%

Table 3: Results for the relativization and time removal tests measured by the BEM metric. We report the intersection between the untransformed (absolute) time references and the two transformations to understand how many of the correct answers are still correct when augmenting the questions to contain, for example, relative time references.

performance. To do so, we remove the temporal references from the questions (Table 3). The model performance decreases by a substantial and surprisingly uniform margin of 38-51%. Conversely, this also means that many temporal questions can be answered (or guessed) correctly without the referenced year's temporal grounding, posing questions about how we evaluate the capture of temporal information in current temporal QA datasets.

What does lower performance mean? We think that discarding dates from a question can result in: (i) the question becoming underspecified and, hence, temporally ambiguous (Piryani et al., 2024). This means that now answers other than the gold answer a may match the question. In general, several different answers may become correct now besides a, as the question can, in principle, refer to any time period. Ideally, LLM should output in this case all the valid answers (or, at least, ask for clarification). In reality, it might just pick one of the answers, likely, the most common one.

(ii) the question becoming more difficult since it is now less informative. The is the case when only one answer a is correct, regardless of the date. A robust LLM should still output the valid answer a, or at least ask for more information.

Case (i) can arise for questions on *common/repeating types of events* or about *highly dynamic facts*. It is also more likely for shorter questions as they are less specific, resulting in more answers having a match. Case (ii) may arise for questions related to *specific events* or *stationary facts*. It is also more likely to happen for more specified questions (where the date is less important as much information is already contained in the question). Given the dynamic nature of the factual questions discussed in our study, we expect decreasing performance after removing temporal references.

4.3 Time Positioning

The time positioning test measures the impact of changing the position of the time information within the question on the models' ability to answer time-scoped questions. Specifically, we rewrite the questions, which usually end with the time reference (i.e., "... in 2019?") to instead begin with this time reference (i.e., "In 2019, ..."). To humans, this rewrite of the question should not make a difference, and similarly, we expect models to be robust to these changes (i.e., no change in performance). The results are shown in Table 4.

Quite remarkably, *all models benefit from time references to be written at the start of the question*, with relative improvements ranging from 16% to 43% in BEM score. While it has been intensively studied that language models mostly focus on the first and the last parts of the input while putting less emphasis on the middle part (Liu et al., 2024), this does not fully explain the observations at hand[6].

4.4 Year Shift

Humans are not always able to remember correct dates. For a question answering, especially over temporal knowledge, to be useful, some lenience regarding errors might be desired. How much exactly is needed and wanted remains to be seen. We

[6] We hypothesize reasons in Appendix G

Model	↔**Positioning**			**Year Shift (num. of years)**					↔**Reversal**		
	Time[end]	Time[front]	Diff.	0	1	5	10	Diff.[0,10]	Fwd	Fwd ∩ Inv	Diff.
`Llama 3.1`	39.2	52.5	↑34%	39.2	29.0	32.9	29.0	↓26%	16.5	6.5	↓61%
`Gemma 2`	40.6	51.8	↑28%	40.6	29.1	33.0	29.1	↓29%	25.1	9.5	↓62%
`Qwen 2.5`	32.9	39.7	↑21%	32.9	18.4	22.2	18.4	↓44%	15.6	4.5	↓71%
`Jamba 1.5`	47.0	**67.2**	↑43%	47.0	35.2	**38.3**	**35.2**	↓25%	35.5	15.9	↓55%
`Cmd-R+`	**47.5**	57.5	↑21%	**47.5**	**33.1**	37.5	33.1	↓30%	29.8	15.0	↓50%
`GPT 4`	**47.5**	55.2	↑**16**%	**47.5**	41.0	31.7	28.1	↓41%	**36.9**	**20.4**	↓**45**%

Table 4: Results of the tests for changing the position of the time reference, the effect of shifting the referenced year, and the temporal reversal test. We report BEM scores.

include this task on how robust models are to corrupting time references by certain amounts. We change the years mentioned in the questions to be wrong by $\{0, 1, 5, 10\}$ years. The results are presented in Table 4. We observe a relatively constant decrease in performance no matter how much we shift the year for all models but `GPT 4`.

4.5 Temporal Reversal

The temporal reversal test is a way to measure the transportability of a fact in another context. Precisely, we test a forward and an inverse formulation of a fact. The forward formulation is the standard question like "Who was the American president in 2005? Answer: Bush". We then reformulate these questions to their inverses, which do not query for the object but for the time of that relation (i.e., "When was Bush the American president? Answer: 2001-2009"). This measures how much the models are susceptible to parroting and how many of these facts are actually understood and usable in differing contexts. The results are shown in Table 4. We notice significant performance drops when looking into questions that were correctly answered in forward *and* inverse forms (45-71%). This suggests that *many of the correct answers are **not** due to a sound temporal understanding of the fact.*

4.6 Temporal Fact Checking

Next, we evaluate the models' ability to judge the factuality of temporal statements. To do so, we use claims that include temporal statements and measure the degree to which the LLMs can generate the ground-truth labels. This is estimating whether a statement is "True," "False," or if there is "Contradicting" information (Table 5).

Given that this task is a three-class classification problem, the results of all models are lacking. Interestingly, we observe lower performance for the most capable `GPT 4`. Upon manual analysis, many models avoid answering questions for their lack of information. This might result from their training, resulting in better-calibrated models and lower performance on this task.

4.7 Event Dating

We next use events from Wikipedia year pages[7] and predict their date in day, month, and year precision. In our experiments and prompt, we specify a format in which we would like to receive the dates ("dd-mm-yyyy"), but we observe many models not adhering to the format. While this is not critical as long as the answer is correct, measuring correctness with the metrics at hand becomes difficult. The recall and contains metrics fail understandably when "11-11-1995" becomes "11th of November 1995". Interestingly, we also observe that BEM and OpenEval are not robust regarding how dates are phrased. We, therefore, use our own date-matching metric that we built using the python dateutil[8] library, which tries to parse a date object from the predicted answers. If the ground-truth date and the parsed date match, the answer is scored with a 1 and otherwise with a 0. After manually inspecting 100 predicted answers for each model, we find 20 different ways to write the dates being used and verify that our metric correctly parses all of them. Using this date-matching metric, we report the event dating performance in Table 5 (middle).

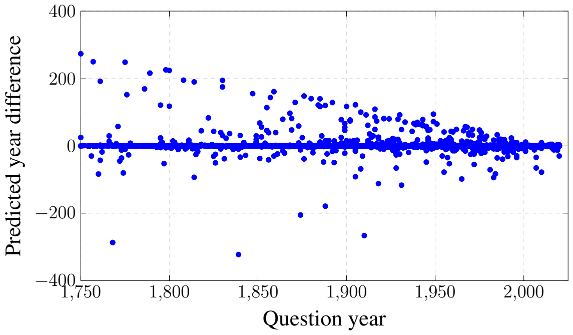

Figure 3: Difference between ground-truth and predicted years for `Llama 3.1`.

As expected, we find the larger models to out-

[7] E.g., https://en.wikipedia.org/wiki/2009

[8] https://github.com/dateutil/dateutil/

Model	↑Fact Checking		↔Event Dating				↔Event Ordering						
	Cont.	OE	Day	Month	Year	Diff.[Y,D]	0	1	5	10	30	100	Diff.[100,0]
`Llama 3.1`	29.1	34.2	9.4	17.7	53.9	↓83%	55.3	55.5	54.9	55.5	54.6	51.8	↓6%
`Gemma 2`	39.9	42.6	20.6	49.9	70.5	↓71%	49.5	57.5	46.9	57.5	50.9	57.5	↓16%
`Qwen 2.5`	**74.7**	31.4	22.8	22.4	62.5	↓64%	46.9	61.7	50.0	61.7	59.3	61.7	↓32%
`Jamba 1.5`	65.5	**52.5**	21.8	30.8	75.4	↓71%	48.5	59.6	49.7	49.6	50.7	49.6	↓**2**%
`Cmd-R+`	46.5	**45.7**	29.4	49.3	68.9	↓57%	66.6	65.0	67.3	70.1	73.2	77.3	↓16%
`GPT 4`	33.1	36.5	**35.9**	**65.3**	**77.0**	↓**53**%	**69.2**	**72.1**	**75.2**	**78.3**	**82.6**	**86.7**	↓25%

Table 5: Results for the fact checking, event dating, and ordering tasks. We report the date match metric for the event dating task and BEM for event ordering. 0,1,5,10,30,100 is the distance in years between the compared events.

Test	Question	Prediction	Consistent /w orig. Prediction
Original	Bernardo Corradi played for which team in 2006?	Fiorentina	
Relativization	Bernardo Corradi played for which team 17 years ago?	Inter Milan	✗
Removal	Bernardo Corradi played for which team?	Italian National Team	✗
Positioning	In 2006, Bernardo Corradi played for which team?	No answer	✗
Reversal	When did Bernardo Corradi play for Fiorentina?	He never did	✗

Table 6: Example of the automatic test suite. For a given question, we automatically create paraphrases inspired by the temporal robustness tests and retrieve answers from the model. Looking into the consistency between the answer to the original question and the test questions can help us judge how much model predictions can be trusted.

perform the smaller models. Also, we observe that *years, as the coarsest granularity, are better captured than months or days*. Going from year-to-day precision, performance drops by 53% or more. The date-matching metric also allows us to measure the difference between predicted and ground-truth dates. We plot the deviation between `Llama 3.1`'s predicted and ground-truth years in Figure 3.

While we observe that most predictions are somewhat close to the actual year (54% even matching it precisely), we also find many questions answered with rather distant years. The worst predicted year was wrong *by over 300 years* (c.f. Figure 3). Interestingly, `Llama 3.1` tends to output too recent dates for many questions.

4.8 Event Ordering

Our last test is whether LLMs can order events chronologically. To do so, we again use the major events from the Wikipedia year pages and always pass two events with the question asking whether A happened before B. This will shed additional light on whether the models have a linear understanding of time and whether this is present, usable information to them. While it has been shown that `Chat-GPT` performs well at event detection and reasoning about causal relationships, it seems not to be proficient at identifying temporal order in the case or discourse analysis (Chan et al., 2024). The results of the event ordering are in Table 5 (right).

First and foremost, we find `Llama 3.1` and `Jamba 1.5` to have consistently bad performance similar to majority classifiers ($\sim 50\%$). For the remaining models, we see the performance increase the further apart the events. Typically, one would expect the models, given they are quite capable of dating events w.r.t. years, but less so w.r.t. days, to perform worse at ordering events from the same year than events that happened 1, 5, or 10 years apart. Even for events from 30 or 100 years apart, where humans are likely to infer the correct order, we only observe a maximum improvement of 6% (`Jamba 1.5`). Unlike humans, who find it natural to differentiate between events that are long time gaps apart, LLMs find it hard to order events despite successfully dating them.

5 Automatic Robustnesss Testing

As with all test suites, our tests for temporal robustness, as discussed in the previous sections, are built a priori and require that we *know the answers* to the questions. However, many of our tests are query-centric reformulations that can be applied to almost generic questions (ending in a temporal reference like "in 2007?"). Thus, we wonder whether we could use these query-centric tests to better judge if the model correctly processed and understood the time component of the question. In this setting, we are *not required* to know the ground-truth answer, which allows us to use this automatic test suite on the fly. This can be beneficial either when hosting the chat/question-answering system to understand when the models might output wrong answers to users or by directly showing the results so that users may use this additional information as contextualization to decide whether they want to trust the model answer or not (example in Table 6).

Given the lack of robustness in the relativization, positioning, and reversal tests, one might question

Model	Llama 3.1	Gemma 2	GPT 4	Avg. Gain
$Q_{No\ Time}$	26.2	22.5	29.3	–
$Q_{+\ Relative}$	27.9	30.1	37.4	+22.6%
$Q_{+\ Absolute}$	40.2	37.0	47.6	+31.4%
$Q_{+\ Time[front]}$	**52.6**	**46.0**	**55.2**	+23.7%

Table 7: We apply the findings of our study into LLM temporal robustness to new questions. By applying automatic transformations to questions in forms to which the LLMs are not robust to (e.g., relative to absolute references), we can improve the QA performance. Values in BEM scores.

any model's temporal literacy. Note that the model might still produce the correct answer, even if the predictions and their consistency might lead us to believe it is not sure about this. However, if the model produces the correct answer, it is more likely due to shortcuts or chance and not because it understood the temporal factual relation.

Next, we show that the query-centric reformulations can be used to estimate whether LLM answers are correct. We use $3,000$ questions for which we have ground-truth answers, creating automatic question reformulations and getting the model predictions for all questions. We evaluate the consistency between the original and test question predictions using the BEM metric, resulting in four values of 0 to 1. We find these four consistency scores to be predictive of the actual correctness of the answer, outperforming a majority classifier on a balanced test set by 14.6% (Llama 3.1). This result emphasizes that besides better understanding and calibrating user trust in the predictions, these *scores can help to evaluate the correctness of model predictions automatically*.

6 Reformulations for Better QA

Lastly, we take another approach to validate the findings from the robustness test. We directly apply the findings about lacking robustness to questions to improve the LLMs for temporal QA. While this was not the main intention of our study, applying the findings emphasizes the usefulness of the study's results. We sample 1k previously unused questions from ArchivalQA and apply the following transformations. First, we remove the time to simulate the effectiveness of adding time references in case they are missing. In subsequent steps, we move from relative to absolute time references and move the temporal reference to the front of the question. The results are shown in Table 7.

Moving from no time reference to relative time offers an average performance boost of 22.6%. Similarly, rewriting relative to absolute references and moving absolute references to the front offer an improvement of 31.4% and 23.7%, respectively. The total improvement of adding absolute time references to the front of questions, which had no time reference, would be 98% on average.

7 Discussion & Conclusion

The temporal robustness tests presented in this work offer a first suite to benchmark temporal processing abilities of LLMs. Besides actionable insights – like avoiding relative time references for most models or starting temporal questions with a time reference – we offer tests that help investigate what temporal understanding is present in models.

The temporal robustness tests may be used in addition to the typical task performance as predeployment checks to evaluate models' abilities and problems better. For example, one might take GPT 4 and Jamba 1.5, which perform very similarly on temporal QA (absolute values in Table 3) and take a closer look at our detailed tests to understand that Jamba 1.5 is more robust to relativization (+11%), but worse at both dating events with day-precision (-14%) and breaks down entirely at ordering event (-20 to -37%).

In this study, we examined the temporal robustness of LLMs. We tested a variety of LLMs using a suite of eight tests assessing different kinds of temporal abilities and robustness to natural paraphrases of questions. While we generally observe higher QA performance for bigger models such as GPT 4, we did not find these models robust to our temporal robustness tests. This study serves as the inaugural benchmark for LLMs' temporal robustness, providing valuable insights into correct temporal information processing and model failures. Further, we found our temporal robustness tests applicable along the entire model lifecycle: 1) For developers to benchmark their models and understand which abilities need improvement. 2) As pre-deployment checks to understand the differences between similarly performing models. 3) By using our automatic tests to help users gauge whether or not to trust the model's predictions. 4) By guiding question reformulations for improved QA performance. We believe this set of tests to help study the temporal robustness of LLMs.

Limitations

Are these tests a comprehensive set? We deal with real-world events, and absolute dates matter; hence, we use several date interventions/perturbations. Temporal ordering and dating are well understood in the literature on events in the Web and IR community (Tran et al., 2015; Campos et al., 2021; Setty et al., 2017; Bradburn, 1999). We leave out operations that belong to temporal algebra (von Karger, 1998) or temporal logic (Konur, 2013) because we are not interested in arbitrary temporal operations and rather focus on events. We are also focused on LLM-based mistakes; our reformulations are natural and not adversarial. Therefore, each transformation is a plausible question, and we leave out all adversarial reformulations. Having made these assumptions (based on the useful and real-world character of the QA task), we acknowledge this is the first step.

What about retrieval? Using a retrieval system and adding additional information is a logical step when striding toward effective QA or chat systems. However, we focus on the innate abilities and temporal robustness of LLMs. When we add context containing the answer to a question, the problem changes from recalling factual information and handling time to a reading comprehension problem – and LLMs are quite proficient at these. The critical part in the retriever setting is retrieving the correct information, which is by no means trivial, especially when discussing historical information that might be relatively rare or incomplete. We deem the retrieval setting to be out of this work's scope.

Ethics Statement

We observe the brittle behavior of LLMs for temporal factual questions. While this may be used to achieve sub-optimal performance, we do not believe this is a directly suitable attack vector to achieve harmful behavior. While not directly derivable from this work, it might be possible to use adversarial attacks to intentionally bias the outputs of LLMs for temporal questions, given the brittle behavior showcased in this study. If successful, this could result in LLMs outputting fake historical information.

References

Vaibhav Adlakha, Parishad BehnamGhader, Xing Han Lu, Nicholas Meade, and Siva Reddy. 2024. Evaluating correctness and faithfulness of instruction-following models for question answering. *Trans. Assoc. Comput. Linguistics*, 12:681–699.

Ashutosh Bajpai, Aaryan Goyal, Atif Anwer, and Tanmoy Chakraborty. 2024. Temporally consistent factuality probing for large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 15864–15881, Miami, Florida, USA. Association for Computational Linguistics.

Himanshu Beniwal, Dishant Patel, Kowsik Nandagopan D, Hritik Ladia, Ankit Yadav, and Mayank Singh. 2024. Remember this event that year? assessing temporal information and understanding in large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 16239–16348, Miami, Florida, USA. Association for Computational Linguistics.

Norman M Bradburn. 1999. Temporal representation and event dating. In *The science of self-report*, pages 61–74. Psychology Press.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Jannis Bulian, Christian Buck, Wojciech Gajewski, Benjamin Börschinger, and Tal Schuster. 2022. Tomayto, tomahto. beyond token-level answer equivalence for question answering evaluation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 291–305. Association for Computational Linguistics.

Ricardo Campos, Gaël Dias, Alípio Mário Jorge, and Adam Jatowt. 2014. Survey of temporal information retrieval and related applications. *ACM Comput. Surv.*, 47(2):15:1–15:41.

Ricardo Campos, Arian Pasquali, Adam Jatowt, Vítor Mangaravite, and Alípio Mário Jorge. 2021. Automatic generation of timelines for past-web events. In *The Past Web: Exploring Web Archives*, pages 225–242. Springer.

Shuyang Cao and Lu Wang. 2022. Time-aware prompting for text generation.

Chunkit Chan, Cheng Jiayang, Weiqi Wang, Yuxin Jiang, Tianqing Fang, Xin Liu, and Yangqiu Song. 2024. Exploring the potential of ChatGPT on sentence level relations: A focus on temporal, causal, and discourse relations. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 684–721, St. Julian's, Malta. Association for Computational Linguistics.

Wenhu Chen, Xinyi Wang, and William Yang Wang. 2021. A dataset for answering time-sensitive questions. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual.*

Zheng Chu, Jingchang Chen, Qianglong Chen, Weijiang Yu, Haotian Wang, Ming Liu, and Bing Qin. 2024. Timebench: A comprehensive evaluation of temporal reasoning abilities in large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 1204–1228. Association for Computational Linguistics.

Jeremy R. Cole, Aditi Chaudhary, Bhuwan Dhingra, and Partha Talukdar. 2023. Salient span masking for temporal understanding. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2023, Dubrovnik, Croatia, May 2-6, 2023*, pages 3044–3052. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

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

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

Mario Giulianelli, Marco Del Tredici, and Raquel Fernández. 2020. Analysing lexical semantic change with contextualised word representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 3960–3973. Association for Computational Linguistics.

Raphael Gruber, Abdelrahman Abdallah, Michael Färber, and Adam Jatowt. 2024. Complextempqa: A large-scale dataset for complex temporal question answering. *CoRR*, abs/2406.04866.

Jai Prakash Gupta, Zhen Qin, Michael Bendersky, and Donald Metzler. 2019. Personalized online spell correction for personal search. In *The World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13-17, 2019*, pages 2785–2791. ACM.

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

Zhen Jia, Abdalghani Abujabal, Rishiraj Saha Roy, Jannik Strötgen, and Gerhard Weikum. 2018. Tempquestions: A benchmark for temporal question answering. In *Companion of the The Web Conference 2018 on The Web Conference 2018, WWW 2018, Lyon , France, April 23-27, 2018*, pages 1057–1062. ACM.

Zhen Jia, Philipp Christmann, and Gerhard Weikum. 2024. TIQ: A benchmark for temporal question answering with implicit time constraints. In *Companion Proceedings of the ACM on Web Conference 2024, WWW 2024, Singapore, Singapore, May 13-17, 2024*, pages 1394–1399. ACM.

Ehsan Kamalloo, Nouha Dziri, Charles Clarke, and Davood Rafiei. 2023. Evaluating open-domain question answering in the era of large language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5591–5606, Toronto, Canada. Association for Computational Linguistics.

Nattiya Kanhabua, Roi Blanco, and Kjetil Nørvåg. 2015. Temporal information retrieval. *Foundations and Trends in Information Retrieval*, 9(2):91–208.

Aobo Kong, Shiwan Zhao, Hao Chen, Qicheng Li, Yong Qin, Ruiqi Sun, Xin Zhou, Enzhi Wang, and Xiaohang Dong. 2024. Better zero-shot reasoning with role-play prompting. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), NAACL 2024, Mexico City, Mexico, June 16-21, 2024*, pages 4099–4113. Association for Computational Linguistics.

Savas Konur. 2013. A survey on temporal logics for specifying and verifying real-time systems. *Frontiers Comput. Sci.*, 7(3):370–403.

Nayeon Lee, Belinda Z. Li, Sinong Wang, Wen-tau Yih, Hao Ma, and Madian Khabsa. 2020. Language models as fact checkers? *CoRR*, abs/2006.04102.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. Lost in the middle: How language models use long contexts. *Trans. Assoc. Comput. Linguistics*, 12:157–173.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 9802–9822. Association for Computational Linguistics.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Seyed Mahed Mousavi, Simone Alghisi, and Giuseppe Riccardi. 2024. Dyknow: Dynamically verifying time-sensitive factual knowledge in llms.

Preslav Nakov, David P. A. Corney, Maram Hasanain, Firoj Alam, Tamer Elsayed, Alberto Barrón-Cedeño, Paolo Papotti, Shaden Shaar, and Giovanni Da San Martino. 2021. Automated fact-checking for assisting human fact-checkers. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021*, pages 4551–4558. ijcai.org.

Bhawna Piryani, Abdelrahman Abdallah, Jamshid Mozafari, and Adam Jatowt. 2024. Detecting temporal ambiguity in questions. In *Findings of the Association for Computational Linguistics: EMNLP 2024, Miami, Florida, USA, November 12-16, 2024*, pages 9620–9634. Association for Computational Linguistics.

Zhen Qin, Zhongliang Li, Michael Bendersky, and Donald Metzler. 2020. Matching cross network for learning to rank in personal search. In *WWW '20: The Web Conference 2020, Taipei, Taiwan, April 20-24, 2020*, pages 2835–2841. ACM / IW3C2.

Wenjun Qiu and Yang Xu. 2022. Histbert: A pre-trained language model for diachronic lexical semantic analysis. *CoRR*, abs/2202.03612.

Guy D. Rosin, Ido Guy, and Kira Radinsky. 2022. Time masking for temporal language models. In *WSDM '22: The Fifteenth ACM International Conference on Web Search and Data Mining, Virtual Event / Tempe, AZ, USA, February 21 - 25, 2022*, pages 833–841. ACM.

Guy D. Rosin and Kira Radinsky. 2022. Temporal attention for language models. In *Findings of the Association for Computational Linguistics: NAACL 2022, Seattle, WA, United States, July 10-15, 2022*, pages 1498–1508. Association for Computational Linguistics.

Apoorv Saxena, Soumen Chakrabarti, and Partha P. Talukdar. 2021. Question answering over temporal knowledge graphs. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 6663–6676. Association for Computational Linguistics.

Vinay Setty, Abhijit Anand, Arunav Mishra, and Avishek Anand. 2017. Modeling event importance for ranking daily news events. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining, WSDM 2017, Cambridge, United Kingdom, February 6-10, 2017*, pages 231–240. ACM.

Giang Binh Tran, Mohammad Alrifai, and Eelco Herder. 2015. Timeline summarization from relevant headlines. In *Advances in Information Retrieval - 37th European Conference on IR Research, ECIR 2015, Vienna, Austria, March 29 - April 2, 2015. Proceedings*, volume 9022 of *Lecture Notes in Computer Science*, pages 245–256.

Venktesh, Abhijit Anand, Avishek Anand, and Vinay Setty. 2024. Quantemp: A real-world open-domain benchmark for fact-checking numerical claims. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2024, Washington DC, USA, July 14-18, 2024*, pages 650–660. ACM.

Burghard von Karger. 1998. Temporal algebra. *Math. Struct. Comput. Sci.*, 8(3):277–320.

Jonas Wallat, Adam Jatowt, and Avishek Anand. 2024. Temporal blind spots in large language models. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining, WSDM 2024, Merida, Mexico, March 4-8, 2024*, pages 683–692. ACM.

Jiexin Wang, Adam Jatowt, Michael Färber, and Masatoshi Yoshikawa. 2021a. Improving question answering for event-focused questions in temporal collections of news articles. *Inf. Retr. J.*, 24(1):29–54.

Jiexin Wang, Adam Jatowt, and Masatoshi Yoshikawa. 2021b. Event occurrence date estimation based on multivariate time series analysis over temporal document collections. In *SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021*, pages 398–407. ACM.

Jiexin Wang, Adam Jatowt, and Masatoshi Yoshikawa. 2022. Archivalqa: A large-scale benchmark dataset for open-domain question answering over historical news collections. In *SIGIR '22: The 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, Madrid, Spain, July 11 - 15, 2022*, pages 3025–3035. ACM.

Jiexin Wang, Adam Jatowt, Masatoshi Yoshikawa, and Yi Cai. 2023. Bitimebert: Extending pre-trained language representations with bi-temporal information. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2023, Taipei, Taiwan, July 23-27, 2023*, pages 812–821. ACM.

Yuqing Wang and Yun Zhao. 2024. TRAM: benchmarking temporal reasoning for large language models. In *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, pages 6389–6415. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Georg Wenzel and Adam Jatowt. 2023. An overview of temporal commonsense reasoning and acquisition. *CoRR*, abs/2308.00002.

Siheng Xiong, Ali Payani, Ramana Kompella, and Faramarz Fekri. 2024. Large language models can learn temporal reasoning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 10452–10470. Association for Computational Linguistics.

Zijun Yao, Yifan Sun, Weicong Ding, Nikhil Rao, and Hui Xiong. 2018. Dynamic word embeddings for evolving semantic discovery. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*. ACM.

Chenhan Yuan, Qianqian Xie, and Sophia Ananiadou. 2023. Zero-shot temporal relation extraction with chatgpt. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks, BioNLP@ACL 2023, Toronto, Canada, 13 July 2023*, pages 92–102. Association for Computational Linguistics.

Chenhan Yuan, Qianqian Xie, Jimin Huang, and Sophia Ananiadou. 2024. Back to the future: Towards explainable temporal reasoning with large language models. In *Proceedings of the ACM on Web Conference 2024, WWW 2024, Singapore, May 13-17, 2024*, pages 1963–1974. ACM.

Hamed Zamani, Michael Bendersky, Xuanhui Wang, and Mingyang Zhang. 2017. Situational context for ranking in personal search. In *Proceedings of the 26th International Conference on World Wide Web, WWW 2017, Perth, Australia, April 3-7, 2017*, pages 1531–1540. ACM.

Ben Zhou, Daniel Khashabi, Qiang Ning, and Dan Roth. 2019. "going on a vacation" takes longer than "going for a walk": A study of temporal commonsense understanding. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 3361–3367. Association for Computational Linguistics.

A General Questions

A.1 Did you describe the limitations of your work?

yes, see Limitations

A.2 Did you discuss any potential risks of your work?

Yes, see Ethics Statement

B Scientific Artifacts

B.1 Did you cite the creators of artifacts you used?

Yes, see Section 3

B.2 Did you discuss the license or terms for use and / or distribution of any artifacts?

1. ArchivalQA: Apache 2.0, `https://github.com/WangJiexin/ArchivalQA`

2. Temporal Facts (Quantemp): CC BY-NC 4.0, `https://github.com/factiverse/QuanTemp`

3. Time-Sensitive QA: BSD 3-Clause, `https://github.com/wenhuchen/Time-Sensitive-QA`

4. Wikipedia Year pages: CC BY-SA 4.0, e.g., `https://en.wikipedia.org/wiki/2006`

We do not plan to distribute these artifacts ourselves but provide scripts to construct the data used in the paper.

B.3 Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?

The used artifacts specify non-commercial use. Our usage was consistent with their specifications.

B.4 Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?

We only collect data from Wikipedia year pages. This may contain names of public figures such as presidents, government figures, or prominent people, and therefore, did not anonymize their names.

B.5 Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?

We cover an overview of the used artifacts in Section 3 as well as Appendix F.1.

B.6 Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created?

We cover an overview of the used artifacts in Section 3 as well as Appendix F.1.

C Computational Experiments

C.1 Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?

The model parameters are listed in Table 8. We did not train or fine-tune models, but ran inference on a larger set of models. Our used infrastructure was a mixture of A100 with 40/80GB memory. Running the entire test suite may take ca. 1 day on one GPU per model, resulting in 6 GPU/days for all models combined.

C.2 Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?

We did not run a hyperparameter search, but we described our experimental setup both in Section 3 and made our code available at `https://anonymous.4open.science/r/temporalrobustness-B3D3/`. The repository contains hyperparameters, prompts, etc.

C.3 Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?

We are reporting single-run results since this study contains descriptive results and does not try to show a clear benefit of using one model over another.

C.4 If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation, such as NLTK, Spacy, ROUGE, etc.), did you report the implementation, model, and parameter settings used?

Details available at: `https://anonymous.4open.science/r/temporalrobustness-B3D3/`

D Human Annotators

D.1 Did you report the full text of instructions given to participants, including, e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?

N/A, did not use human annotators

D.2 Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?

N/A, did not use human annotators

D.3 Did you discuss whether and how consent was obtained from people whose data you're using/curating?

N/A, did not use human annotators

D.4 Was the data collection protocol approved (or determined exempt) by an ethics review board?

N/A, did not use human annotators

D.5 Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?

N/A, did not use human annotators

E Use of AI Assistants

E.1 Did you include information about your use of AI assistants?

We did not use AI assistants to generate text or perform research directly. We did use ChatGPT and Grammarly to perform reformulations of existing text as well as to fix grammatical errors.

F Additional Setup Details

F.1 Extended Discussion of Temporal Source Datasets

Time-sensitive-QA dataset (Chen et al., 2021) is constructed by mining time-evolving facts from WikiData and aligning them to their corresponding Wikipedia pages, employing crowd workers to verify and calibrate noisy facts, and generating question-answer pairs based on the annotated time-sensitive facts. The dataset contains 40,000 question-answer pairs focusing on around 5,500 time-evolving facts; it is structured into two variants based on difficulty: easy and hard.

TemporalQuestions dataset (Wang et al., 2021a) is designed to evaluate the capability of QA systems to handle time-scoped questions. This dataset focuses on questions related to specific events and their temporal aspects, derived from historical news archives and other temporally rich sources. The dataset contains 1,000 human-generated questions about major events, half of which are explicitly and half implicitly time-scoped, meaning half of the questions contain temporal expressions. In contrast, the remaining ones lack any temporal references.

ArchivalQA (Wang et al., 2022) is a large-scale collection designed specifically for temporal news QA, containing 532,444 question-answer pairs, often on detailed and minor aspects. These pairs are derived from the New York Times Annotated Corpus, which spans news articles published between January 1, 1987, and June 19, 2007. The dataset-constructing framework with automatic question generation and filtering steps ensures high-quality and non-ambiguous questions.

F.2 Additional Model Details

As shown in Table 8, we use a selection of competitive LLMs. Specifically, we use thee following versions: `Llama 3.1`[9], `Gemma 2`[10], `Jamba 1.5`[11], `Qwen 2.5`[12], `Cmd-R+`[13], and `GPT 4`[14].

F.2.1 Temporal QA Performance

We evaluate the temporal robustness of LLMs in the study. Still, for completeness reasons, we also provide the downstream QA performance of our models on three established temporal QA benchmark datasets. The results are shown in Table 9. The performance of all models, on all metrics, leaves an opportunity for improvement.

F.2.2 Effect of Prompts

We note that most of our study uses the standard prompts and did not include any prompt engineering or established best practice (e.g., Chain-of-Thought (Wei et al., 2022) or role-playing[15] (Kong et al., 2024)) prompts. We also experiment with these two best practice prompts[16] and show the results for a selection of models in Table 10. While the historian role-playing prompt performs competitively across the board, the CoT prompt does not and might be unsuitable for factual recall, which usually might not involve multi-step reasoning. Lastly, we expect prompt tuning to improve the overall model performance. Still, we did not see clear evidence that better-performing models consistently outperform others in robustness to our temporal paraphrases.

[9] https://huggingface.co/meta-llama/Llama-3.1-8B-Instruct

[10] https://huggingface.co/google/gemma-2-27b-it

[11] https://huggingface.co/ai21labs/AI21-Jamba-1.5-Mini

[12] https://huggingface.co/Qwen/Qwen2.5-32B-Instruct

[13] https://huggingface.co/CohereForAI/c4ai-command-r-plus-4bit

[14] https://platform.openai.com/docs/models/gp#gpt-4-turbo-and-gpt-4

[15] "You are a historian [...]"

[16] All prompts will be made available with our code.

Model Name	Mode Size	Notes	Cutoff
Llama 3.1	8B	Instruction-tuned version	Dec. 2023
Jamba 1.5	12B active, 52B total	Mixture-of-Experts model that combines mamba (state-space) and transformer blocks. 8bit quant.	Mar. 2024
Gemma 2	27B	Instruction-tuned version	Jun. 2024
Qwen 2.5	32B	Instruction-tuned version	2023
Cmd-R+	104B params	RAG-optimized language model, weights openly available. Uses 4-bit quantization	N.S.
GPT 4	N.S.	OpenAI's flagship GPT model (gpt-4-1106-preview)	Apr. 2023

Table 8: Summary of different models with their respective details

Model	ArchivalQA				TemporalQuestions				Time-Sensitive-QA			
	Recall	Cont.	BEM	OE	Recall	Cont.	BEM	OE	Recall	Cont.	BEM	OE
Llama 3.1	22.2	18.5	41.4	23.2	63.1	58.9	76.8	62.1	13.3	7.7	30.7	13.7
Jamba 1.5	30.0	24.3	**49.9**	35.4	73.3	68.6	**86.9**	74.2	28.6	18.2	44.7	32.3
Gemma 2	29.3	25.0	40.6	32.3	76.8	72.2	85.7	78.4	23.4	16.0	32.4	26.0
Qwen 2.5	25.6	21.1	31.6	26.0	63.6	60.1	72.1	64.4	19.5	10.0	18.6	16.5
Cmd-R+	31.8	26.1	46.5	37.0	76.3	72.1	81.5	79.1	30.7	21.1	38.6	32.7
GPT 4	**38.5**	**32.8**	46.3	**39.8**	**81.7**	**76.3**	86.7	**81.9**	**44.2**	**33.2**	**46.3**	**39.3**

Table 9: Performance of our models on common temporal factual QA benchmarks.

G More discussion on the Effect of Positioning Time

Given that we employ a system prompt and a prompt that specifically asks for the following question to be answered, the time reference at the front of the question is hardly at the beginning of the model input. However, the time reference at the end is almost at the end of the input and should, therefore, be focussed on by the model. Yet, the performance is found to be superior when the time reference is before the question. We hypothesize that a different thing is at play here: The residual stream does not have enough bandwidth to store all historical information on certain entities and relations. Meng et al. (2022) found that when answering factual questions about entities, the embeddings of the last entity token would be enriched with as much information as possible on that entity by retrieving it from the feed-forward layers. This information is then copied to the last token embedding, where the attention mechanism selects the information necessary to answer the question from the embedding.

Let us look at an example in our normal question form: *Who was the American president in 2019?* Remember that all LLMs in this study are autoregressive language models (i.e., their attention may only look at the previous context of a given token). We see that the token "president," in which the factual information will be aggregated, has no "understanding" that it needs to find information on what was the case in 2019. Therefore, it would either save the most recent information or try to aggregate all information. Using the most recent information will likely fail with historical knowledge (and given our other results, we do not believe this to be the case). Trying to enrich the embedding with all information on the American president might fail because there is too much information. When the last token's attention then tries to retrieve the correct information about our year, it might not be accessible, and the question might be answered incorrectly. However, if the question starts with the time–reference, the entity token ("president") can be precisely enriched with the information from the correct years and information from other years may be then de-prioritized or discarded. While this hypothesis needs to be thoroughly tested, for our case of measuring temporal robustness, we can conclude that the desired output would be models that are robust to where the time references occur in the question. When aiming for the best QA performance, however, this result suggests formulating temporal questions to start with their temporal references.

Model	**Prompt**	**ArchivalQA**				**TemporalQuestions**				**Time-Sensitive hard QA**			
		Recall	Cont.	BEM	OE	Recall	Cont.	BEM	OE	Recall	Cont.	BEM	OE
Cmd-R+	default	31.8	**26.1**	46.5	**37.0**	**76.3**	**72.1**	81.5	**79.1**	30.7	21.1	38.6	32.7
	CoT	12.9	12.2	20.5	35.2	34.3	30.9	43.7	71.0	13.6	8.3	16.8	**43.8**
	historian	**32.5**	**26.1**	46.4	36.0	74.7	69.7	**82.2**	77.2	**32.5**	**21.9**	**38.8**	33.6
Jamba 1.5	default	30.0	24.3	**49.9**	35.4	73.3	68.6	86.9	74.2	28.6	18.2	44.7	32.3
	CoT	29.9	24.4	49.7	35.1	73.2	68.5	**87.6**	72.5	28.5	18.3	45.5	32.9
	historian	**30.6**	**25.0**	47.5	**36.1**	**74.7**	**69.6**	83.9	**76.7**	**30.3**	**20.0**	**46.4**	**34.5**
Llama 3.1	default	22.2	18.5	**41.4**	23.2	63.1	58.9	**76.8**	62.1	13.3	7.7	**30.7**	13.7
	CoT	5.2	3.3	10.3	**36.1**	11.6	8.9	18.8	**71.8**	7.9	3.3	11.9	**33.8**
	historian	**23.0**	**19.2**	41.1	23.6	**64.4**	**60.1**	76.2	60.9	**15.5**	**9.8**	29.9	15.1

Table 10: Overview of the temporal QA of our models when using different prompting schemes.