Around the World in 24 Hours
Probing LLM Knowledge of Time and Place

Carolin Holtermann[1], Paul Röttger[2], Anne Lauscher[1]
[1]University of Hamburg, Germany
[2]University of Bocconi, Italy

carolin.holtermann@uni-hamburg.de

Abstract

Reasoning over time and space is essential for understanding our world. However, the abilities of language models in this area are largely unexplored as previous work has tested their abilities for logical reasoning in terms of time and space in isolation or only in simple or artificial environments. In this paper, we present the first evaluation of the ability of language models to *jointly* reason over time and space. To enable our analysis, we create GEOTEMP, a dataset of 320k prompts covering 289 cities in 217 countries and 37 time zones. Using GEOTEMP, we evaluate eight open chat models from three model families for different combinations of temporal and geographic knowledge. We find that most models perform well on reasoning tasks involving only temporal knowledge and that overall performance improves with scale. However, performance remains poor in tasks that require connecting temporal and geographical information. We do not find clear correlations of performance with specific geographic regions. Instead, we find a significant performance increase for location names with low model perplexity, suggesting their repeated occurrence during model training. We further demonstrate that model performance is heavily influenced by prompt formulation – a direct injection of geographical knowledge leads to performance gains, whereas, surprisingly, techniques like chain-of-thought prompting decrease performance on simpler tasks.[1]

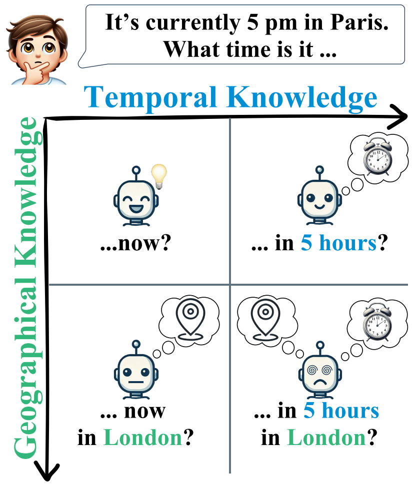


Figure 1: **Combinations of temporal and geographic knowledge we test with GEOTEMP.** Models struggle most with jointly reasoning over space and time.

1 Introduction

> *"Stop worrying about the world ending today. It's already tomorrow in Australia."*
> — Charles M. Schulz

Human civilization has developed systems and conventions to organize life and foster community, such as a standardized global time and calendar system, time zones based on geographic location, and daylight saving time to align activities across different regions (Hayden, 1987). These conventions, deeply rooted in our understanding of the world, play a crucial role in decision-making, especially in today's globalized working environment. As technology advances, large language models (LLMs) will increasingly be used to simplify and optimize tasks (Eloundou et al., 2023), for example in the logistical planning of cross-border shipping or the optimal planning of business trips. To reliably perform such tasks, they must be able to incorporate temporal and geographical information into their reasoning process. Prior work has evaluated the temporal reasoning ability of LLMs (e.g. Dhingra et al., 2022; Fatemi et al., 2024) as well as their spatial understanding of the world (e.g. Gurnee and Tegmark, 2024; Moayeri et al., 2024). However, little work has been done on evaluating LLM ability to reason *jointly* over space and time, outside of simple and artificial settings such as multiple-choice questions (Wang and Zhao, 2024)

[1]We release all data and code at https://github.com/UhhDS/GeoTemp.

Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 22875–22897
July 27 - August 1, 2025

or synthetic data (Fatemi et al., 2024).

In this paper, we address this gap by analyzing the ability of LLMs to combine geographical and temporal knowledge in reasoning tasks, using global time zones as a testbed. To this end, we present GEOTEMP, which contains 320k question prompts covering 289 locations in 37 time zones. GEOTEMP is explicitly designed to evaluate LLM across different levels of temporal and geographical reasoning (Figure 1). Using GEOTEMP we answer the following research questions:

1) How well do LLMs reason about time, place, and their combination? We test eight state-of-the-art chat-optimized LLMs on four different tasks requiring different levels of temporal and/or geographical knowledge (§5). We find that most models are able to perform simple calculations of time, but fail as soon as they have to perform additional geographical reasoning. Although we see improvements with model scale, the best-performing model is only 25.4% accurate on questions requiring both temporal and geographical knowledge.

2) What factors influence model performance on geotemporal tasks? In contrast to prior work on geographic knowledge (Moayeri et al., 2024), we do not find clear performance disparities across geographic regions (§6). However, we do observe a significant correlation between model performance and location name perplexities. Direct probing of time zones reveals that models generally possess knowledge of individual geographical and temporal facts, but are unable to combine the two effectively, which limits performance on geotemporal tasks.

3) Can model performance be improved by injecting geotemporal knowledge? We provide models with location-specific time zones to guide their reasoning (§7), and find significant performance improvements. However, surprisingly, these improvements are mostly for locations for which, in our prior probing, we found the models to know the time zone of already. This suggests that, while basic geotemporal facts may be already present in the models, this knowledge requires activation to solve more complex geotemporal tasks.

2 Related Work

Testing temporal knowledge Several studies benchmark the temporal knowledge of LLMs. The TimeBank corpus (Pustejovsky et al., 2003) focuses on annotating news articles for temporal expressions, events, and their links. Recent work expands to analyze temporal knowledge beyond event extraction, with benchmarks like McTaco (Zhou et al., 2019), TempLama (Dhingra et al., 2022), or SituatedQA (Zhang and Choi, 2021) probing models' factual knowledge of time-varying. Other studies explore their temporal reasoning capabilities, including relations between temporal expressions (Thukral et al., 2021), event ordering (Tan et al., 2023; Wei et al., 2023; Xiong et al., 2024), and duration (Vashishtha et al., 2020), and more complex tasks like multi-hop reasoning (Tan et al., 2024) or video reasoning (Liao et al., 2024). Recent work also examines the temporal knowledge alignment of LLMs to given timestamps (Kasai et al., 2024; Liska et al., 2022; Zhao et al., 2024). Finally, Nylund et al. (2024) and Gurnee and Tegmark (2024) analyze temporal representations within models through probing and methods from task arithmetic.

Testing geographical knowledge Previous work on geographic knowledge in LLMs focuses on the prediction of geolocations (Godey et al., 2024), their internal model representation (Gurnee and Tegmark, 2024), and how this knowledge can be adapted (Liétard et al., 2021). Other works extend these efforts by evaluating the ability of LLMs to recall information about different countries and examine model-intrinsic and location-specific biases (Moayeri et al., 2024; Manvi et al., 2024b,a).

Testing geotemporal knowledge Two recent works partially address geotemporal reasoning but provide limited insight into the ability of models to combine geographical and temporal knowledge. 1) The TRAM corpus (Wang and Zhao, 2024) evaluates temporal reasoning, including time zone calculations as a subtask. However, it focuses on model performance and does not provide a deeper analysis of the model's ability to combine temporal and geographical knowledge. Moreover, its multiple-choice format does not allow open-ended responses and thus strongly guides the model in its response. 2) The TOT benchmark (Fatemi et al., 2024) is a fully synthetic data set that tests temporal reasoning without including real-world location information. It evaluates time zone calculations, but based on the provision of time zones, not geographic locations.

3 GEOTEMP

We introduce GEOTEMP, the first probing dataset specifically designed to evaluate the geotempo-

Task Name	Template
VERIFICATION	*What time is it now in* l_1*?*
TIMETIME	*What time is it in x hours?*
TIMEPLACE	*What time is it now in* l_2*?*
TIMETIMEPLACE	*What time is it in x hours in* l_2*?*

Table 1: **Overview of tasks and their templates** we use to construct test prompts in GEOTEMP. We prepend to each template information about the current date, time, and location as follows: *Today is {Time&Date} in {*l_1*}.*

ral knowledge of LLMs. By carefully constructing synthetic prompts that progressively require more complex geotemporal reasoning, GEOTEMP allows us to isolate failure points while minimizing the risk of using exact prompts seen during training. However, grounded in real-world LLM usage, GEOTEMP combines temporal and geographical knowledge, using global time zones as its basis.

3.1 Dataset Creation

We create GEOTEMP in three steps: 1) We collect time zones along with cities located in them; 2) We craft four task templates representing varying combinations of geotemporal knowledge, coupling each of these tasks with one or two of our locations, depending on the task template; 3) We compose the final dataset by sampling representative task-location combinations from above, and embedding them into the task templates.

1) Collecting time zones and locations We use the Olson Time Zone Database (OTZD) to compile a diverse selection of locations across countries and time zones.[2] The OTZD contains time zone information for 596 representative locations worldwide, including rules for daylight saving time. It is actively maintained by the non-profit organization ICANN, ensuring reliability and accuracy.[3]

The 596 timezone names we extract from the OTZD typically follow a naming convention of *Area/City* (e.g. *Europe/Madrid*). To ensure that all entries in our dataset refer to precise locations, we manually exclude time zones that represent entire regions, such as *Brazil/West*, as well as outdated time zones that are no longer in use. After applying these filters, we are left with a refined set of 460 distinct and precise locations. For each of these locations, we gather additional *city*-level data, including population, latitude, and longitude, using the Opendatasoft API, to enable more comprehensive analysis later on.[4]

2) Crafting task templates Next, we define four tasks requiring different combinations of geotemporal knowledge. We present these in Table 1.

VERIFICATION provides the model with a date and time in a given location l_1, and asks for that same time again (e.g. *Today is Monday, June 19th at 10:33 AM in 2023 in Novokuznetsk, Russia. What time is it now in Novokuznetsk, Russia?*). This allows us to assess the model's basic comprehension of the task and its ability to respond with a correctly formatted time.

TIMETIME provides a date and time and a specific location l_1, then asks what the time will be in that same location in a specified number of hours (e.g. *What time is it in 3 hours?*). To accomplish this task, the model only needs to perform basic temporal reasoning, whereas geographical reasoning is irrelevant. This template is adapted from prior work on LLM temporal knowledge and reasoning (Dhingra et al., 2022; Tan et al., 2024).

TIMEPLACE gives the model the current date and time at a specified location l_1 and asks for the current time at another given location l_2. To solve this task, the model requires knowledge of the time zones at each location and must determine the resulting time zone difference.

TIMETIMEPLACE provides the model with a date and time at a specific location l_1, then asks the model for the time in another location l_2 but in a specific number of hours. Solving this requires the model to consider the time shift before or after the time zone change, in addition to TIMEPLACE. This tests the model's ability to handle both temporal shifts and geographic knowledge simultaneously.

3) Composing the test set With GEOTEMP, we want to cover a wide variety of locations while avoiding over-representing any particular combination of locations. To do this, we select a representative sample from the 460 locations obtained from OTZD, such that all 37 UTC time zones and 217 ISO country codes are included. This results in a refined selection of 289 locations. Next, we generate the Cartesian product of all selected locations $l_1 \times l_2$ to cover all possible combinations within the TIMEPLACE and TIMETIMEPLACE templates. Finally, we construct all evaluation prompts of GEOTEMP

[2] https://www.iana.org/time-zones
[3] https://www.icann.org/
[4] https://public.opendatasoft.com/explore/dataset/

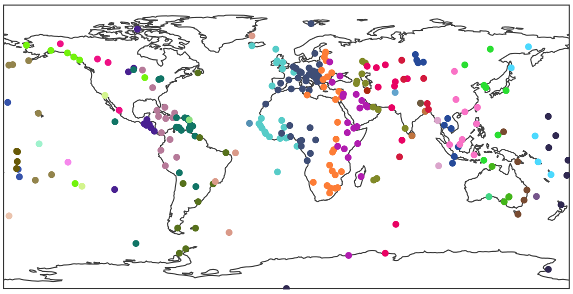


Figure 2: **Geographical distribution of all 289 locations in GEOTEMP.** Color indicates UTC time zone.

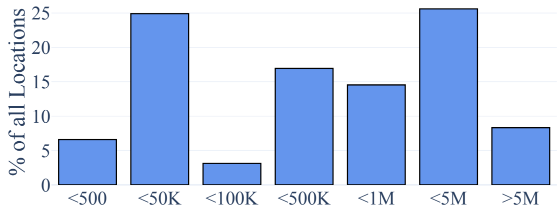


Figure 3: **Distribution of cities included in GEOTEMP according to their number of inhabitants.** For each bin, we show the percentage of locations in our dataset that belong to this bin.

by applying every task template of Table 1 to each $l_1 \times l_2$ location combinations. To avoid confusion with city names that exist in multiple countries (e.g. London in the US vs. London in the UK), we include the country information alongside the city name in the prompt. For each input prompt, we select the time and date at random within the year 2023, i.e. a recent year, to avoid outdated references and automatically calculate the target time and date which serves as the ground truth label for our evaluation algorithm. This results in a final set of 332,928 test prompts.

3.2 Dataset Analysis

GEOTEMP covers a diverse set of locations and time zones across all continents, as shown in Figure 2, including Antarctica and several islands. GEOTEMP also represents smaller and therefore often less well-known cities, such as "Atikokan" in Canada. The distribution of the number of inhabitants per city is shown in Figure 3. More than 6% of the locations in our dataset have $\leq$500 inhabitants, and 50% of the cities have $\leq$500k inhabitants. For a detailed statistical analysis of the location information in GEOTEMP, see Appendix A.

4 Overall Experimental Setup

4.1 Models and Inference

We test eight openly accessible chat-optimized LLMs spanning three model families and different model sizes on GEOTEMP. All models have shown competitive performance on the LMSyS Leaderboard.[5] Specifically, we evaluate Llama2-Chat (Touvron et al., 2023) in the model sizes 7B, 13B and 70B, Llama3-Instruct (AI@Meta, 2024) in both available model sizes (8B and 70B) and Qwen2-Instruct (Yang et al., 2024) in the model sizes 1.5B, 7B and 72B. We run all models on 1-4 Nvidia A6000 GPUs using the `simplegen` python library (Attanasio, 2023). In all experiments, we use a maximum length of 256 and a temperature of 0 to make model responses deterministic.

During inference, we run each prompt of GEOTEMP using three instruction templates to analyze their effect on model performance. The first instruction type (*neutral*) is shown in Table 1 and provides no additional information on how the model should answer the question. This leaves the model the ability to choose its own explanation methodology. The second type appends the suffix *"Think step by step."*, asking the model to apply *chain-of-thought* (*CoT*) reasoning. Lastly, the third type appends the suffix *"Keep your answer short and just answer with the correct time and date"*, which encourages concise responses (*short*).

4.2 Evaluation Protocol: Regex

We use a custom matching algorithm based on regex patterns to extract the time and date from the model responses. For example, from a response like *"In 5 hours it is 02:17 AM"*, we extract *"02:17 AM"* and compare it to the gold standard answer. We chose to parse the model's response rather than enforce a strict output format. This allows us to keep prompts more natural and ensures that we do not penalize correct answers that only slightly deviate from the expected structure.

We develop and test the regex evaluator in two steps. First, we ask two independent annotators to evaluate a sample of 3,600 model responses for their correctness (*true / false*) compared to the gold answers. Second, we split this data 50:50 into a development set, for constructing and optimizing regex patterns and a test set for evaluation. We achieve an almost perfect Cohen's κ annotator

[5] `https://huggingface.co/spaces/lmsys/chatbot-arena-leaderboard`

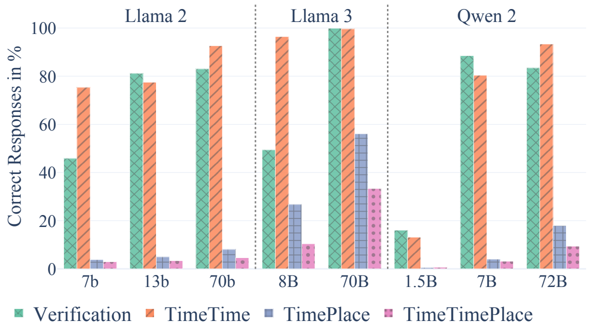


Figure 4: **Model accuracy on GEOTEMP.** We show the percent of correct model responses (obtained via majority vote across different instruction types) for different models for our four geotemporal tasks (VERIFICATION, TIMETIME, TIMEPLACE, TIMETIMEPLACE).

agreement on the labeling task according to the scale of Landis and Koch (1977), with agreements in the range of 90.8-99.2% across models, ensuring the quality of the obtained labels. Our regex evaluator achieves an accuracy of at least 98 % for each model on the test set portion of the annotated subset. These scores confirm that we can use our regex evaluator for our experiments without introducing substantial noise into the evaluation results. Note also that the regex, with near-perfect accuracy, is vastly more efficient than training a response classifier or using LLM-as-a-judge. For further details on the validation process, see Appendix B.

5 How Well Do LLMs Reason About Time, Place, and their Combination?

For a reliable performance estimate on our dataset, we show aggregated results across all three instruction types: We use the majority vote per input instance, i.e., the answer obtained for at least two corresponding instructions. This procedure thus also reflects possible improvements through chain-of-thought prompting on more complex tasks. We present these combined results in Figure 4 and show detailed results in Appendix C.

5.1 Overall Results

Performance varies drastically across models and tasks. Generally, while the VERIFICATION and TIMETIME tasks are solvable for most models, they struggle when asked to incorporate location-specific knowledge. For instance, Llama3-70B yields the best results with 56.1% of correctly answered prompts for the TIMEPLACE category and 33.4% correct answers for the TIMETIMEPLACE category. Interestingly, even though one would expect a monotonous performance decrease with higher task complexity, as for Llama2-13B and the smaller Qwen2 models, all other models seem to struggle with the VERIFICATION task while performing much better on the TIMETIME task. We hypothesize this may be linked to ongoing training on mathematical problems, as LLMs have generally been shown to struggle with such tasks (Dziri et al., 2024). Very simple or even rhetorical questions, such as our VERIFICATION question, are probably less frequently included in the training data as they do not contain a common task to be solved. An analysis of several model responses shows that the models often attempt to read into the question a task that is not actually posed, such as providing further current times in other time zones.

5.2 Model Scaling

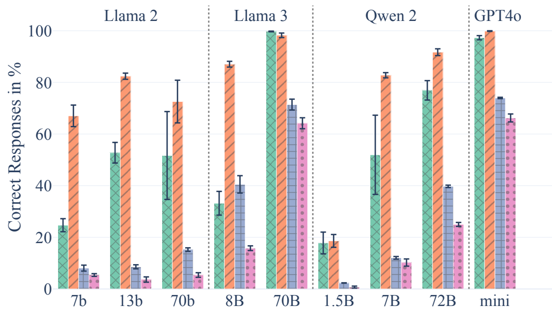
While we see consistently poor performance for all models on the two tasks that require geographical knowledge (TIMEPLACE and TIMETIMEPLACE), we also observe consistent scaling effects. For Llama2, this effect is more apparent for the two other and less complex tasks, while for Llama3 and Qwen2, a performance increase can be observed for all four tasks as the model size increases. In the 70B version, Llama3 thus achieves almost 100% accuracy for the VERIFICATION and TIMETIME task and a performance increase of 29.3% for TIMEPLACE and 23.0% on TIMETIMEPLACE compared to Llama3-8B. For Qwen2, however, an increase in model size results in only marginal improvements, with final performances of 18.0% and 9.4% on the complex tasks, respectively. For Llama2, the gains are even less apparent. This raises doubts as to whether the models will understand the underlying mechanisms for solving these tasks simply by upscaling.

5.3 Instruction Types

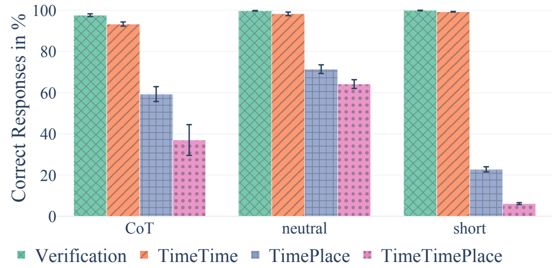
Previously, we analyzed aggregated results across the three instruction types (*neutral*, *chain-of-thought*, *short*), but examining them individually also reveals interesting patterns. Forcing concise answers (*short*) leads to decreased performance on complex tasks across all models, while improving results on simpler tasks, except for Qwen2. In contrast, *chain-of-thought* instructions result in significantly worse performance on simpler tasks for Llama2 and slightly worse for Llama3 compared to *neutral*, particularly for Llama2-70B in the VERIFICATION task. Qualitative analysis of the model's answers suggests that the model is trying to solve a more difficult task than requested and ultimately gets stuck in its own explanations. These results are related to the work of Sprague et al. (2024), who show that *chain-of-thought* prompting primarily benefits math-related tasks. This highlights the need for a more comprehensive analysis of the effect of different instruction types on questions of varying difficulty, which we leave for future work. We provide detailed results in the Appendix C.

5.4 Robustness

Since the exact prompt formulation can have a strong impact on performance (Sclar et al., 2024), we additionally perform a robustness analysis. For each instruction type, we randomly sample 1,000 instances and generate results for two additional prompts with slight variations in wording. To enable comparison with state-of-the-art closed models, we also conduct the same experiment using `gpt4o-mini`. In Figure 5a, we show the mean and standard deviation across all models for the instruction type *neutral*, while Figure 5b presents detailed results for all instruction types obtained with Llama3-70B. Results for the remaining models are provided in Appendix C. While we observe significant performance fluctuations for some models and tasks, importantly, the overall trends between the models, their sizes, and instruction types are stable. Encouragingly, the error for the TIMEPLACE and TIMETIMEPLACE tasks, which we mainly analyze, is very low, highlighting the robustness of our findings.



(a) Robustness results for the *neutral* instruction type.



(b) Llama3-70B results across instruction types.

Figure 5: **Robustness analysis.** We show the mean and standard deviation of model answers across three different prompt variations. In (a), the results for the *neutral* instruction type across all models. In (b) we show the results obtained by Llama3-70B for each instruction type (*neutral*, *chain-of-thought*, *short*).

6 What Factors Influence Model Performance?

6.1 Country Biases

Following Moayeri et al. (2024), we investigate performance differences based on location, comparing Western vs. Non-Western countries, varying income levels, and population densities. For this, we focus on the tasks requiring location-specific knowledge (TIMEPLACE and TIMETIMEPLACE), aggregating results by start and target country. Figure 6 depicts the response accuracy aggregated by the target country of Llama3-70B. We provide results for the aggregation by start country in Appendix C.2, which show no significant differences.

Llama3-70B performs best on questions that include African countries. Similarly, the model performs well for cities in South America and around Russia. For cities in North America, Oceania, Antarctica, and Greenland, on the other hand, we see significantly worse results. This is contrary to our hypothesis that models develop a performance bias towards Western countries. Further aggregations at the continental level, population size, and income level did not reveal any clear patterns in this direction either.

6.2 Location Name Perplexity

Since we could not identify a clear location bias in the models, we explored other factors that might influence their performance.

Detailed setup We hypothesize that the familiarity of the models with certain locations leads to them performing better in a task. Since we cannot access the models' training data, we make use of the models' perplexity as a proxy to assess their familiarity with specific locations. To this end, we use a simple and static sentence template: *"I live in {city}, {country}"*, replace the placeholders with the corresponding city and country information for each location l from GEOTEMP and compute the perplexity for each model. To better interpret the perplexity levels, we categorize the resulting

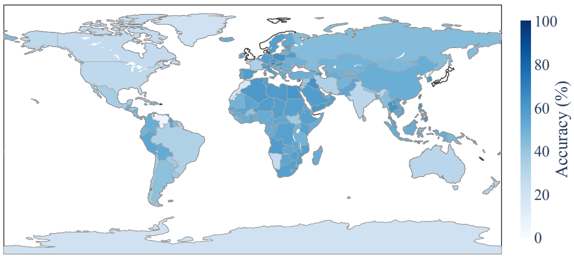


Figure 6: **Model accuracy aggregated by target country** for the tasks TIMEPLACE and TIMETIMEPLACE. We neither observe biases towards Western nor towards higher-income or population-dense countries.

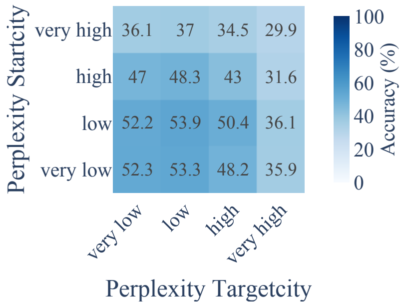


Figure 7: **Effect of location perplexities.** We show the performance of Llama3-70B on the TIMEPLACE and TIMETIMEPLACE task aggregated by the perplexity of the start location and target location.

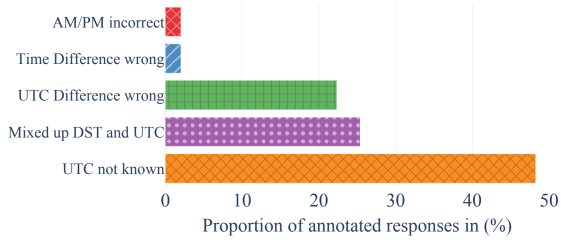


Figure 8: **Results of our error analysis.** We show the proportion of model errors (in %) per error category.

perplexity scores into four bins. Specifically, for each model, we determine the perplexity distribution over all locations in our dataset and classify locations with a perplexity score within the 25% quantile of this distribution as *"very low"* perplexity, those up to the 50% quantile as *"low"*, up to the 75% quantile as *"high"* and the remaining values as *"very high"*. We then associate these with the corresponding start and target locations in our dataset and compute the average model performance in percent of correct answers for each combination of perplexity scores of the two locations in question. Note that, again, we are limiting this analysis to the TIMEPLACE and TIMETIMEPLACE tasks, where location information is relevant.

Results We focus on the result of Llama3-70B, shown in Figure 7, because of its strong performance across tasks. However, we find that these findings also hold for most other models which we show in the Appendix C.3. For Llama3 we observe a gradual decrease in accuracy from *low* to *very high* perplexity locations, with a surprisingly high performance drop between the combination of *low x low* (53.9) and *very high x very high* (29.9%) of 22.5%. This trend is also visible for Qwen2-72B with a performance difference of 15.5% between the combinations *very low x very low* and *very high x very high*. Our findings thus suggest that model performance is biased in favor of locations that likely appear more frequently in model training data, rather than being biased *per se* in the direction of Western countries, as one might have expected.

6.3 Error Analysis

To get more precise insights into why models fail to combine temporal and geographical knowledge, we perform a manual error analysis for the best-performing model, Llama3-70B. We randomly sample 200 of the model's incorrect responses on the TIMETIMEPLACE task. One author then annotated this sample, assigning errors to one of five error categories (Figure 8). We find that 25.3% of errors are attributable to the model's failure to appropriately handle the conversion between DST and UTC for the current time. 22.3% of errors result from the fact that, despite correct UTC knowledge for both locations, the time difference was not calculated correctly. However, 48.2% of the errors are due to the fact that the model does not use the correct UTC in its decision-making process for at least one of the locations.

6.4 Location-Specific Knowledge

We seek to further investigate why models perform so poorly on tasks requiring a combination of temporal and geographical knowledge. From our TIMETIME task, it is evident that models are generally able to perform basic time calculations, hence we look at the time zone knowledge of the models for the locations. To do this, we prompt the models to provide us with the corresponding time zone for each location in our dataset. For an easier parsing of the results, we additionally force the model to respond following the scheme: *"City: UTC±X"*. The outcomes of this evaluation, presented in Table 2, are surprising in light of the previous results: All

Model	Size	UTC Known (%)
Llama2	7B	70.9
	13B	71.3
	70B	80.9
Llama3	8B	84.1
	70B	**90.0**
Qwen2	1.5B	39.3
	7B	65.4
	72B	86.3

Table 2: **Knowledge about time zones**. We present the proportion of locations where models assigned the correct time zone (in % of all locations in our dataset) when explicitly prompted to do so.

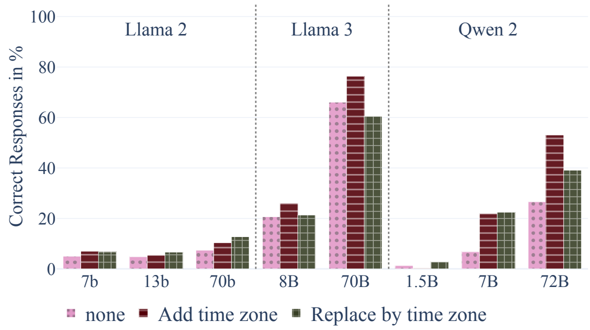


Figure 9: **Results of UTC knowledge injection** on the TIMETIMEPLACE task. Colors respond to different amounts of knowledge injected.

models can predict the correct UTC timezone for at least 65% of the locations, except for Qwen2-1.5B with only 39.3%. For Llama3-70B, it is as much as 90%. Consequently, the poor results from our main experiments (Figure 4) cannot be attributed solely to a lack of knowledge about the cities and their geographical locations. Rather, models fail to make use of this knowledge when not explicitly asked to provide it.

7 Can Model Performance be Improved by Injecting Geotemporal Knowledge?

Detailed setup So far, we found that the models are generally able to perform temporal calculations and that knowledge about the individual locations is present, yet they fail when combining the two. In a final experiment, we now examine whether explicitly injecting geographical knowledge by providing time zone information helps to activate the required knowledge (and thus, improve performance on the most complex task TIMETIMEPLACE). We examine two different setups: In the first setup, we add the respective time zone information of the city in addition to the input prompt (*"add time zone"*). In the second variant, we omit the city names from the prompt and replace them entirely with the time zone information (*"replace by time zone"*).

Results We present model performance with injected information in comparison to prior performance on the TIMETIMEPLACE task in Figure 9. We see that the injection of geotemporal knowledge improves performance across models. For Llama2, however, improvements are limited, pointing to a general lack of understanding of the concept of time zones. For the large versions of the other models, we see a significantly larger effect. Llama3-70B achieves a performance of 76.3% by adding the time zone information, and Qwen2-72B of 53%. We also find that models do not perform as well when city information is replaced with time zone information as when time zone information is added on top. This suggests that models do draw on their geographical knowledge to solve this task.

8 Discussion

Current language models struggle to perform geotemporal reasoning. Our dataset and analysis allowed us to identify reasons for this. First, we find that the only model with fewer than 7 billion parameters in our test (Qwen2-1.5B) is largely ineffective for geotemporal reasoning as required by the tasks in our dataset. In contrast, we show that bigger models can indeed perform basic temporal calculations, indicating a solid understanding of temporal concepts like the 24-hour format and AM/PM notation. Especially, the example of Llama3-70B, with almost 100% accuracy on this task (TEMPTEMP), shows that targeted training (e.g., high-quality data and specific training for STEM tasks) can achieve a significant performance improvement. Second, most models – except for Qwen2-1.5B – correctly determine time zones for at least 65% of the locations when explicitly asked to do so, demonstrating general geographic competencies. Third, despite this, their performance on tasks combining temporal and geographical knowledge is poor. Our qualitative analysis reveals that when models have to perform complex tasks that require them to access and combine different types of knowledge simultaneously, they are no longer able to retrieve time zone information that they knew in isolation and instead start hallucinating. Our knowledge injection tests support this finding, as reducing task complexity by limiting the amount of information to be retrieved at once significantly improves model performance. This analysis points to a broader issue: **models may often possess the necessary knowledge but fail to retrieve and combine it effectively for complex tasks** (and steering techniques like *chain-of-thought* may only partially help). Looking into the future, we also believe that simply increasing the model size will not solve the problem. Instead, improvements towards more systematic and step-wise knowledge retrieval or the use of tools may be necessary to prevent hallucinations and sustainably enhance performance on complex reasoning tasks.

9 Conclusion

In this article, we introduced GEOTEMP, the first benchmark designed to evaluate the ability of LLMs to reason over both temporal and geographical knowledge at different combinations of knowledge. GEOTEMP consists of over 320k prompts covering 289 locations across 37 timezones. Using our dataset, we analyzed eight open chat-optimized LLMs across varying sizes and found that, while models perform adequately on tasks involving only temporal knowledge, their performance significantly drops when they are required to combine temporal and geographical information. Our findings indicate that this limitation persists also at larger model scales, suggesting that even the best models today struggle with more complex reasoning tasks involving the interplay of time and space. We also observed that the models perform better for locations where they exhibit low perplexity, which may indicate that prior exposure to certain geographic locations during training helps.

Limitations

Regex Evaluation We use a self-developed algorithm to evaluate the open-ended model responses, which inevitably causes some degree of noise. This was necessary as a full human evaluation for a dataset of this size would not be feasible, and LLM as a judge would be too resource-intensive and also not fully reliable. However, we are conducting a comprehensive validation of our methodology, providing very high accuracy, which leads us to believe that we can consider the noise to be low. We are therefore confident that our results remain valid.

Prompt Robustness Evaluation on Selected Subset Due to resource constraints, we conduct robustness checks only on a subset of our dataset. It is therefore possible that the true uncertainties around our results are larger. Third, in our experiments, we evaluate models for which we unfortunately do not have any precise information about their pretraining data. To make a statement about the model's familiarity with certain terms, we thus use model perplexity as a proxy for the frequency of the term's occurrence in the training data. Yet, this metric is not fully reliable and may introduce inaccuracies. A future evaluation of fully open models and their training data could provide additional insights.

No Tasks Involving Location Alone GEOTEMP does not cover purely location-based tasks, as such a task would closely resemble existing work and might not significantly advance the field. Furthermore, a task that involves predicting a location is less conducive to leveraging temporal information compared to the task we have chosen to focus on.

Acknowledgements

Paul Röttger is a member of the Data and Marketing Insights research unit of the Bocconi Institute for Data Science and Analysis, and is supported by a MUR FARE 2020 initiative under grant agreement Prot. R20YSMBZ8S (INDOMITA). The work of Carolin Holtermann and Anne Lauscher is funded by the Excellence Strategy of the German Federal Government and the Federal States.

References

AI@Meta. 2024. Llama 3 model card.

Giuseppe Attanasio. 2023. Simple Generation. https://github.com/MilaNLProc/simple-generation.

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.

Nouha Dziri, Ximing Lu, Melanie Sclar, Xiang Lorraine Li, Liwei Jiang, Bill Yuchen Lin, Peter West, Chandra Bhagavatula, Ronan Le Bras, Jena D. Hwang, Soumya Sanyal, Sean Welleck, Xiang Ren, Allyson Ettinger, Zaid Harchaoui, and Yejin Choi. 2024. Faith and fate: limits of transformers on compositionality. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

Tyna Eloundou, Sam Manning, Pamela Mishkin, and Daniel Rock. 2023. Gpts are gpts: An early look at the labor market impact potential of large language models. *Preprint*, arXiv:2303.10130.

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. *Preprint*, arXiv:2406.09170.

Nathan Godey, Éric de la Clergerie, and Benoît Sagot. 2024. On the scaling laws of geographical representation in language models. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 12416–12422, Torino, Italia. ELRA and ICCL.

Wes Gurnee and Max Tegmark. 2024. Language models represent space and time. *Preprint*, arXiv:2310.02207.

F. Gregory Hayden. 1987. Evolution of time constructs and their impact on socioeconomic planning. *Journal of Economic Issues*, 21(3):1281–1312.

Jungo Kasai, Keisuke Sakaguchi, Yoichi Takahashi, Ronan Le Bras, Akari Asai, Xinyan Velocity Yu, Dragomir Radev, Noah A. Smith, Yejin Choi, and Kentaro Inui. 2024. Realtime qa: what's the answer right now? In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

J. Richard Landis and Gary G. Koch. 1977. The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174.

Ruotong Liao, Max Erler, Huiyu Wang, Guangyao Zhai, Gengyuan Zhang, Yunpu Ma, and Volker Tresp. 2024. Videoinsta: Zero-shot long video understanding via informative spatial-temporal reasoning with llms.

Bastien Liétard, Mostafa Abdou, and Anders Søgaard. 2021. Do language models know the way to Rome? In *Proceedings of the Fourth BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 510–517, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Adam Liska, Tomas Kocisky, Elena Gribovskaya, Tayfun Terzi, Eren Sezener, Devang Agrawal, Cyprien De Masson D'Autume, Tim Scholtes, Manzil Zaheer, Susannah Young, Ellen Gilsenan-Mcmahon, Sophia Austin, Phil Blunsom, and Angeliki Lazaridou. 2022. StreamingQA: A benchmark for adaptation to new knowledge over time in question answering models. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 13604–13622. PMLR.

Rohin Manvi, Samar Khanna, Marshall Burke, David Lobell, and Stefano Ermon. 2024a. Large language models are geographically biased. *Preprint*, arXiv:2402.02680.

Rohin Manvi, Samar Khanna, Gengchen Mai, Marshall Burke, David Lobell, and Stefano Ermon. 2024b. Geollm: Extracting geospatial knowledge from large language models. *Preprint*, arXiv:2310.06213.

Mazda Moayeri, Elham Tabassi, and Soheil Feizi. 2024. Worldbench: Quantifying geographic disparities in llm factual recall. In *Proceedings of the 2024 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '24, page 1211–1228, New York, NY, USA. Association for Computing Machinery.

Kai Nylund, Suchin Gururangan, and Noah Smith. 2024. Time is encoded in the weights of finetuned language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2571–2587, Bangkok, Thailand. Association for Computational Linguistics.

James Pustejovsky, Patrick Hanks, Roser Saurí, Andrew See, Rob Gaizauskas, Andrea Setzer, Dragomir Radev, Beth Sundheim, David Day, Lisa Ferro, and Marcia Lazo. 2003. The timebank corpus. *Proceedings of Corpus Linguistics*.

Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. 2024. Quantifying language models' sensitivity to spurious features in prompt design or: How i learned to start worrying about prompt formatting. In *The Twelfth International Conference on Learning Representations*.

Zayne Sprague, Fangcong Yin, Juan Diego Rodriguez, Dongwei Jiang, Manya Wadhwa, Prasann Singhal, Xinyu Zhao, Xi Ye, Kyle Mahowald, and Greg Durrett. 2024. To cot or not to cot? chain-of-thought helps mainly on math and symbolic reasoning. *Preprint*, arXiv:2409.12183.

Qingyu Tan, Hwee Tou Ng, and Lidong Bing. 2023. Towards benchmarking and improving the temporal reasoning capability of large language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14820–14835, Toronto, Canada. Association for Computational Linguistics.

Qingyu Tan, Hwee Tou Ng, and Lidong Bing. 2024. Towards robust temporal reasoning of large language models via a multi-hop QA dataset and pseudo-instruction tuning. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 6272–6286, Bangkok, Thailand and virtual meeting. Association for Computational Linguistics.

Shivin Thukral, Kunal Kukreja, and Christian Kavouras. 2021. Probing language models for understanding of temporal expressions. In *Proceedings of the Fourth BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 396–406, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. *Preprint*, arXiv:2307.09288.

Siddharth Vashishtha, Adam Poliak, Yash Kumar Lal, Benjamin Van Durme, and Aaron Steven White. 2020. Temporal reasoning in natural language inference. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4070–4078, Online. Association for Computational Linguistics.

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

Yifan Wei, Yisong Su, Huanhuan Ma, Xiaoyan Yu, Fangyu Lei, Yuanzhe Zhang, Jun Zhao, and Kang Liu. 2023. MenatQA: A new dataset for testing the temporal comprehension and reasoning abilities of large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 1434–1447, Singapore. Association for Computational Linguistics.

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)*, pages 10452–10470, Bangkok, Thailand. Association for Computational Linguistics.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jianxin Yang, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Xuejing Liu, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, Zhifang Guo, and Zhihao Fan. 2024. Qwen2 technical report. *Preprint*, arXiv:2407.10671.

Michael Zhang and Eunsol Choi. 2021. SituatedQA: Incorporating extra-linguistic contexts into QA. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7371–7387, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Bowen Zhao, Zander Brumbaugh, Yizhong Wang, Hannaneh Hajishirzi, and Noah A. Smith. 2024. Set the clock: Temporal alignment of pretrained language models. *Preprint*, arXiv:2402.16797.

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)*, pages 3363–3369, Hong Kong, China. Association for Computational Linguistics.

Appendix

Table of Contents

A Dataset

A.1 Sources

We present a list of datasets and frameworks that we use for creating GeoTemp. The timezone information is obtained from the OTZD, which we access using the Python libraries `pytz` and `datetime`.

Purpose	Name	Source	License
Datasets	Country population, Country names, Income Level, Regions	Worldbank https://data.worldbank.org/	CC BY 4.0
	City population	Geonames/Opendatasoft https://public.opendatasoft.com/explore/dataset/geonames-all-cities-with-a-population-500/api/?disjunctive.country	CC BY 4.0
Frameworks	pytz==2024.1 python-dateutil==2.8.2	https://pypi.org/project/pytz/ https://docs.python.org/3/library/datetime.html	

Table 3: Overview of datasets and frameworks used in our work.

A.2 Quantitative Analysis

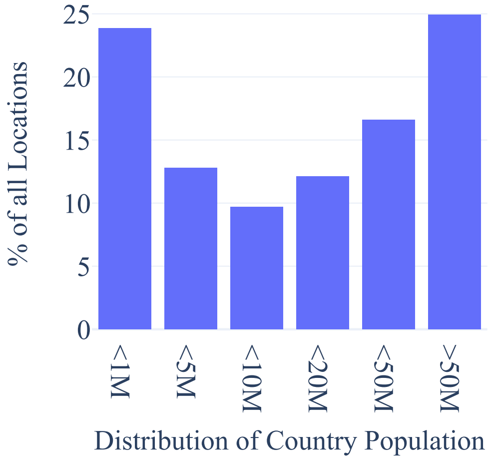


Figure 10: Histogram Country Population

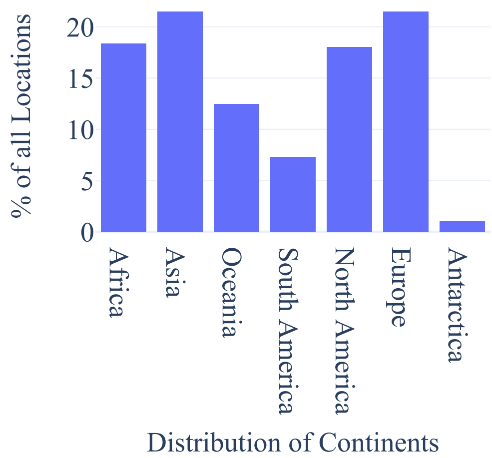


Figure 12: Histogram Continents of Locations

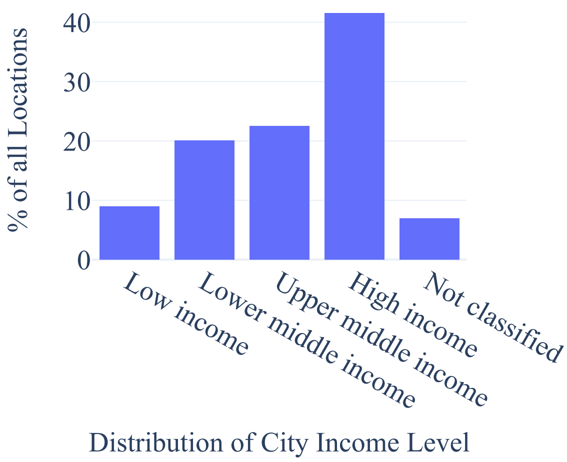


Figure 11: Histogram Country Income Level

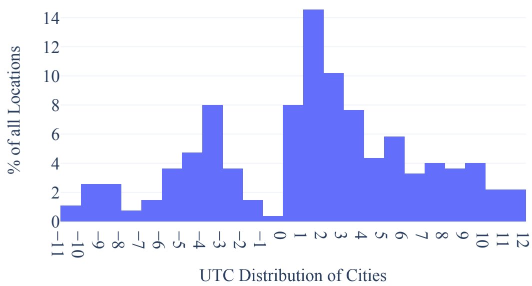


Figure 13: Histogram UTC of Locations

B Evaluation Setup

Annotation We have two independent human annotators label this set of answers for their correctness with regard to the target answer given by the *pytz* package. Both annotators are fluent in English and are familiar with the task at hand. Since the annotators both worked in the authors' research environment, they did not receive any compensation. They were also informed about how their annotations would be used.

Expectedly, we achieve an overall agreement of 96.4% of Cohen's κ on the annotation set, constituting a very high agreement between the annotators and thus very reliable labels. In a qualitative analysis, it can be seen that most disagreement was caused by either random errors or on the `Verification` task. This is because models are often confused by this rather rhetorical type of question, and instead of returning the answer, they convert the time at a location into the corresponding time in UTC. This is generally not a wrong answer, but we specify for the annotation guidelines that the model must also respond with the correct time at the respective location. To obtain the final label, disagreements between the annotators were overruled by an expert decision judgment.

Model	Size	Cohen's κ
Llama2	7B	90.8
	13B	94.3
	70B	97.5
Llama3	8B	97.5
	70B	99.2
Qwen2	1.5B	96.9
	7B	97.4
	72B	97.4

Table 4: **Inter-annotator agreement** on a sample of 3,600 model answers from GeoTemp. We present the Cohen's κ between the two annotators in percent.

Validation Next, we split the annotated subset using a 50:50 split into a calibration and a test portion. Note that, since we are not actually training a new model but rather using the set to calibrate our own matching algorithm, a larger test portion to validate the generalization of our approach on unseen data is more important than the calibration portion. The evaluation results for our matching algorithm compared to the human annotations are shown in Table 5.

The results show, that we can achieve an accuracy of at least 98% on the test set portions for each model using our algorithm. The lowest precision score of 94.7% was achieved for the 1.5B version of Qwen2. This is due to the fact that the model provides very few correct answers, and thus the true positive rate is very low, causing the precision score to decrease rapidly even though there is only one misclassified instance. Our validation results therefore suggest that our evaluation method delivers reliable results for the succeeding analysis.

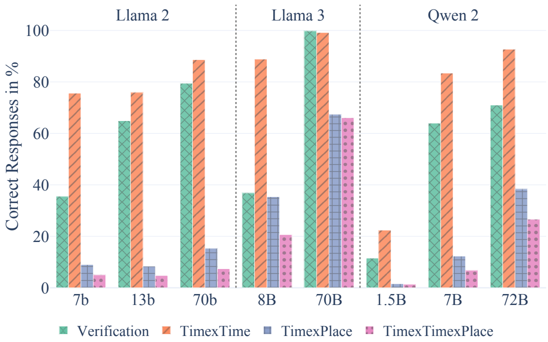
Model	Size	Acc	P	R
Llama2	7B	99.5	98.4	100.0
	13B	98.0	95.9	98.6
	70B	98.0	98.9	96.7
Llama3	8B	98.5	97.8	98.9
	70B	98.5	100.0	98.2
Qwen2	1.5B	99.5	94.7	100.0
	7B	98.0	97.5	97.5
	72B	99.0	100.0	98.3

Table 5: **Performance of our regex evaluator.** Evaluation results on the sample test set of our dataset. We present the accuracy, precision, and recall for each model in percent on an annotated subset of 100 prompts each.

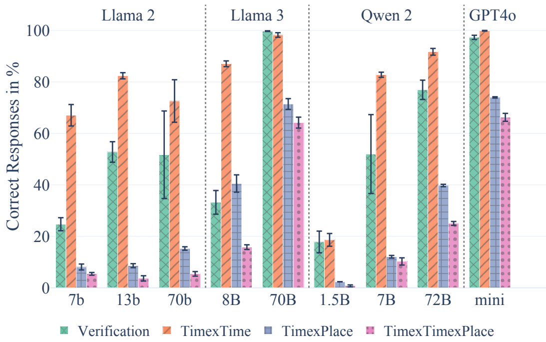
C Experimental Results

C.1 Robustness Analysis

We present the accuracy results on the whole dataset and the robustness results on a subset of 1000 prompts with three different prompt variations across the three different instruction types: (1) neutral, (2) short, and (3) CoT. We prompt GPT-4O-MINI-2024-07-18 in the same way as the other models, which we detailed in Section 4.1 via the OpenAI API[6].

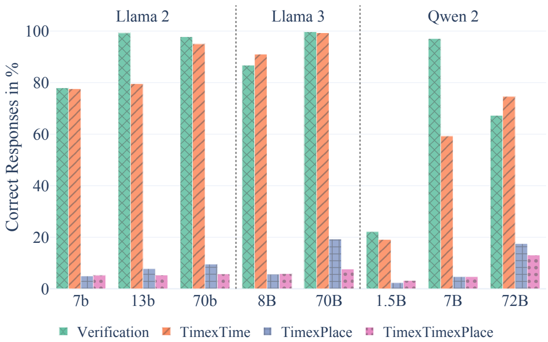


(a) Overall Results

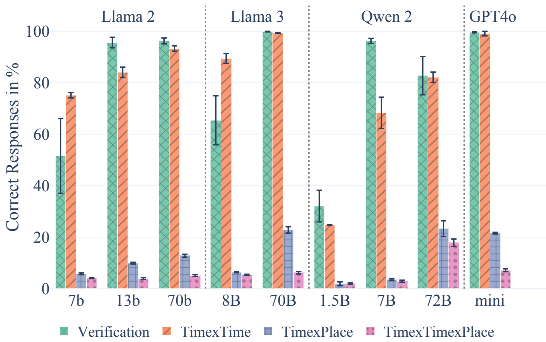


(b) Robustnesschecks

Figure 14: Model Accuracy neutral prompt

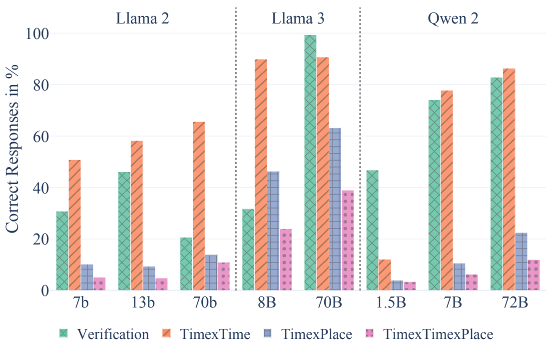


(a) Overall Results

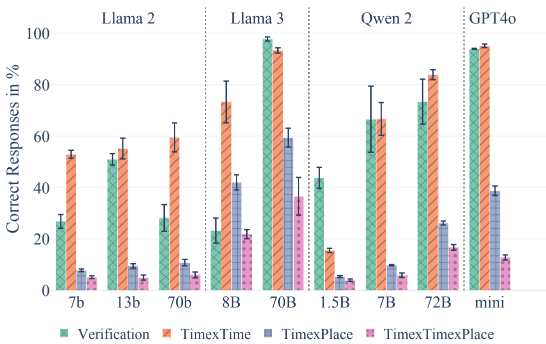


(b) Robustnesschecks

Figure 15: Model Accuracy short prompt



(a) Overall Results



(b) Robustnesschecks

Figure 16: Model Accuracy CoT prompt

[6]https://platform.openai.com/docs/overview

C.2 Country Bias Analysis

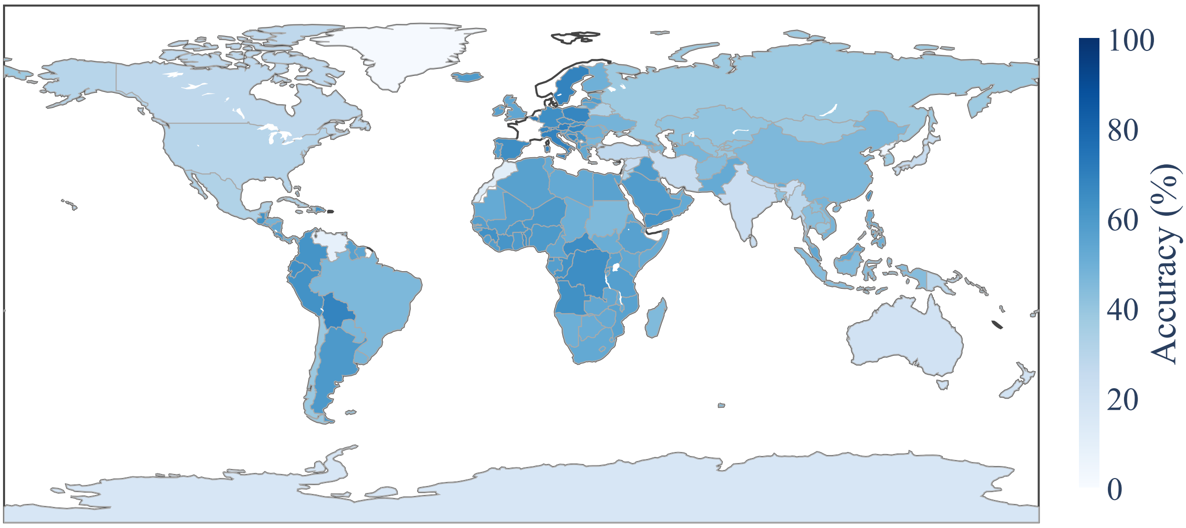
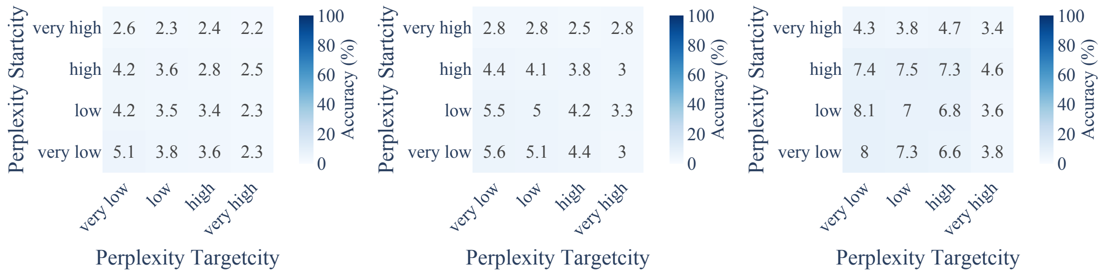


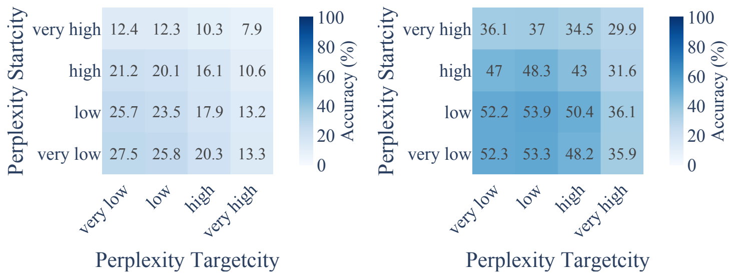
Figure 17: **Model accuracy aggregated by start country** for the tasks TIMEPLACE and TIMETIMEPLACE. We neither observe biases towards Western nor towards higher-income or population-dense countries.

C.3 Perplexity Analysis



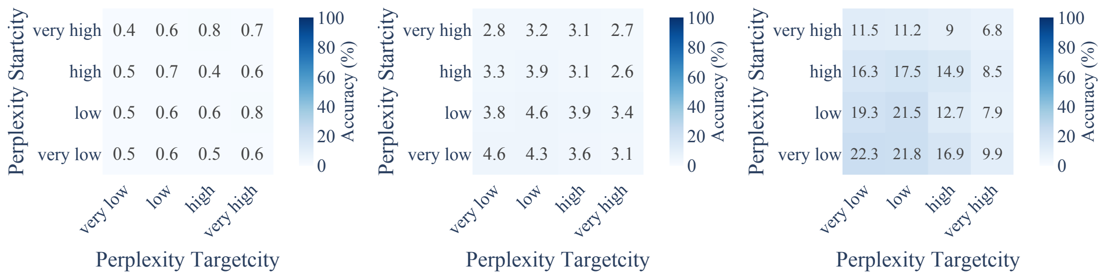
(a) Llama2 7B (b) Llama2 13B (c) Llama2 70B

Figure 18: Llama2 accuracy distribution for different City Perplexities



(a) Llama3 8B (b) Llama3 70B

Figure 19: Llama3 accuracy distribution for different City Perplexities



(a) Qwen2 1.5B (b) Qwen2 7B (c) Qwen2 72B

Figure 20: Qwen2 accuracy distribution for different City Perplexities

D Qualitative Model Responses

Qualitative Examples of Model Responses We present examples of model responses obtained for the three different instruction types: (1) short, (2) neutral, and (3) CoT, respectively.

Model	Task	Input Prompt	Answer	Model Response	Label
Qwen2-1.5B	VERIFICATION	Today is Friday, January 6th at 09:28 AM in 2023 in Valletta, Malta. What time is it now in Valletta, Malta? [...]	2023-01-06 09:28 AM	user 10:47 PM	✗
Qwen2-7B	TIMETIMEPLACE	Today is Sunday, June 18th at 09:34 PM in 2023 in Mazatlan, Mexico. What time is it in 4 hours in Luxembourg, Luxembourg? [...]	2023-06-19 10:34 AM	Assistant: Wednesday, June 22nd, 2023, 01:34 AM in Luxembourg, Luxembourg.	✗
Qwen2-7B	TIMETIME	Today is Monday, December 4th at 02:19 AM in 2023 in Nome, United States. What time is it in 7 hours? [...]	2023-12-04 09:19 AM	answer Tuesday, December 4th at 09:19 AM in 2023	✓
Llama3-8B	TIMETIMEPLACE	Today is Friday, February 3rd at 06:36 AM in 2023 in Gambier Islands, French Polynesia. What time is it in 2 hours in Manila, Philippines? [...]	2023-02-04 01:36 AM	assistant February 3rd, 2023, 08:36 PM	✗
Llama3-8B	TIMETIME	Today is Tuesday, May 23rd at 05:32 AM in 2023 in Dublin, Ireland. What time is it in 2 hours? [...]	2023-05-23 07:32 AM	assistant 07:32 AM, May 23rd, 2023	✓
Llama3-70B	TIMEPLACE	Today is Thursday, September 14th at 05:24 PM in 2023 in Zagreb, Croatia. What time is it now in Managua, Nicaragua? [...]	2023-09-14 09:24 AM	assistant September 14th, 2023, 11:24 AM	✗
Llama3-70B	TIMETIME	Today is Sunday, June 4th at 03:27 AM in 2023 in Aden, Yemen. What time is it in 4 hours? [...]	2023-06-04 07:27 AM	assistant Sunday, June 4th, 07:27 AM	✓
Llama2-7B	TIMETIMEPLACE	Today is Saturday, October 21st at 07:47 PM in 2023 in Rothera Point, Adelaide Island. What time is it in 4 hours in Tbilisi, Georgia? [...]	2023-10-22 06:47 AM	In 4 hours from 07:47 PM in Rothera Point, Adelaide Island on October 21st, 2023, it will be 11:47 PM in Tbilisi, Georgia on October 21st, 2023.	✗
Llama2-13B	VERIFICATION	Today is Saturday, December 2nd at 01:06 AM in 2023 in Astrakhan, Russia. What time is it now in Astrakhan, Russia? [...]	2023-12-02 01:06 AM	Sure! It is currently Saturday, December 2nd at 01:06 AM in Astrakhan, Russia.	✓
Llama2-70B	TIMETIME	Today is Tuesday, July 18th at 02:09 AM in 2023 in Cocos, Australia. What time is it in 2 hours? [...]	2023-07-18 04:09 AM	Sure! In 2 hours, it will be: 04:09 AM on Tuesday, July 18th, 2023.	✓
Llama2-70B	TIMETIMEPLACE	Today is Sunday, February 5th at 12:56 AM in 2023 in Alofi, Niue. What time is it in 7 hours in Brisbane, Australia? [...]	2023-02-06 04:56 AM	Sure! In 7 hours, it will be 7:56 AM on Sunday, February 5th, 2023 in Brisbane, Australia.	✗

Table 6: Examples of the model responses on the short instruction template.

Model	Task	Input Prompt	Answer	Model Response	Label
Qwen2-1.5B	TIMEPLACE	Today is Saturday, October 28th at 11:50 PM in 2023 in Vilnius, Lithuania. What time is it now in Chita, Russia?	2023-10-29 05:50 AM	assistant It is currently 9:50 AM on October 29th, 2023 in Chita, Russia.	✗
Qwen2-7B	TIMETIME	Today is Tuesday, June 27th at 01:36 PM in 2023 in Grand Turk, Turks and Caicos Islands. What time is it in 8 hours?	2023-06-27 09:36 PM	answer To calculate the time 8 hours from now from 01:36 PM on Tuesday, June 27th, 2023, in Grand Turk, Turks and Caicos Islands, we simply add 8 hours to this time. 01:36 PM + 8 hours = 09:36 PM Therefore, 8 hours from now, it will be 09:36 PM on the same day, Tuesday, June 27th, 2023, in Grand Turk, Turks and Caicos Islands.	✓
Qwen2-72B	VERIFICATION	Today is Thursday, August 10th at 05:04 PM in 2023 in Novokuznetsk, Russia. What time is it now in Novokuznetsk, Russia?	2023-08-10 05:04 PM	[Incomplete Generation] :04 PM on Thursday, August 10th, 2023 in Novokuznetsk, Russia. However, if you're asking for the current time and it's after that date and time, you would need to adjust for the time that has passed since then. Since I can't provide real-time information, you should check the current time using a reliable time source or a device that updates in real-time.	✗
Llama3-8B	TIMETIME-PLACE	Today is Friday, August 11th at 08:52 AM in 2023 in Ljubljana, Slovenia. What time is it in 9 hours in Palmer Station, Anvers Island?	2023-08-11 12:52 PM	assistant A nice geography and time zone question! Ljubljana, Slovenia is in the Central European Summer Time (CEST) zone, which is UTC+2. Palmer Station, Anvers Island is in the Antarctic Time (ANT) zone, which is UTC-3. To find the time in Palmer Station, we need to subtract 5 hours from the current time in Ljubljana. Since it's 08:52 AM in Ljubljana, we subtract 5 hours to get: 08:52 AM - 5 hours = 03:52 AM So, it would be 03:52 AM in Palmer Station, Anvers Island.	✗
Llama3-8B	TIMETIME	Today is Thursday, July 27th at 02:32 PM in 2023 in Brisbane, Australia. What time is it in 7 hours?	2023-07-27 09:32 PM	assistant Since it's 02:32 PM in Brisbane, Australia, which is in the Australian Eastern Standard Time (AEST) zone, it's already past the standard time zone's 12:00 PM (noon) mark. In 7 hours, it would be: 02:32 PM + 7 hours = 09:32 PM So, in 7 hours, it would be 09:32 PM on Thursday, July 27th, 2023, in Brisbane, Australia.	✓
Llama3-70B	VERIFICATION	Today is Thursday, February 2nd at 12:36 PM in 2023 in Johannesburg, South Africa. What time is it now in Johannesburg, South Africa?	2023-02-02 12:36 PM	assistant That's an easy one! According to the information you provided, the current time in Johannesburg, South Africa is 12:36 PM on Thursday, February 2nd, 2023.	✓
Llama2-7B	TIMEPLACE	Today is Sunday, July 16th at 01:08 PM in 2023 in Broken Hill, Australia. What time is it now in Rarotonga, Cook Islands?	2023-07-15 05:38 PM	To find out what time it is in Rarotonga, Cook Islands, we need to consider the time difference between Broken Hill, Australia and Rarotonga. Rarotonga is located in the Cook Islands, which are located in the Pacific Ocean, about 3,000 miles (4,800 km) south of Broken Hill, Australia. Rarotonga is 18 hours ahead of Broken Hill. So, if it is 01:08 PM (13:08) on July 16th in Broken Hill, Australia, it would be 6:08 AM on July 16th in Rarotonga, Cook Islands (UTC+13).	✗
Llama2-13B	TIMEPLACE	Today is Thursday, September 14th at 05:45 PM in 2023 in Tunis, Tunisia. What time is it now in Sydney, Australia?	2023-09-15 02:45 AM	Sydney, Australia is 10 hours ahead of Tunis, Tunisia. So, if it is 05:45 PM in Tunis on Thursday, September 14th, 2023, it would be 11:45 PM in Sydney on Thursday, September 14th, 2023.	✗
Llama2-70B	TIMEPLACE	Today is Wednesday, January 11th at 08:58 AM in 2023 in Qostanay, Kazakhstan. What time is it now in Tehran, Iran?	2023-01-11 06:28 AM	Tehran, Iran is in the Iran Standard Time (IRST) time zone, which is 3 hours and 30 minutes ahead of Qostanay, Kazakhstan. So, if it is 08:58 AM in Qostanay, it would be 12:28 PM in Tehran.	✗

Table 7: Examples of the model responses on the neutral instruction template.

Model	Task	Input Prompt	Answer	Model Response	Label
Qwen2-7B	TIMETIME	Today is Thursday, September 21st at 01:07 PM in 2023 in Caracas, Venezuela. What time is it in 2 hours? Think step by step.	2023-09-21 03:07 PM	user To find out what time it will be in 2 hours from now in Caracas, Venezuela, given that it's currently Thursday, September 21st at 01:07 PM in 2023, follow these steps: 1. **Add the hours to the current time**: - Current time: 01:07 PM - Adding 2 hours: 01:07 PM + 2 hours = 03:07 PM So, in 2 hours from now, it will be 03:07 PM on Thursday, September 21st, 2023, in Caracas, Venezuela. 2. **Check for any time zone adjustments**: - Caracas uses UTC-4 (or UTC-5 during Daylight Saving Time). However, since we're only adding 2 hours and not considering Daylight Saving Time changes, there's no adjustment needed for this calculation. Therefore, the final answer is that in 2 hours from now, it will be 03:07 PM on Thursday, September 21st, 2023, in Caracas, Venezuela.	✓
Qwen2-72B	TIMETIME-PLACE	Today is Sunday, January 8th at 03:12 PM in 2023 in Lisbon, Portugal. What time is it in 9 hours in Noronha, Brazil? Think step by step.	2023-01-08 10:12 PM	Fernando de Noronha, Brazil is 2 hours behind Lisbon, Portugal. If it's 03:12 PM on Sunday, January 8th in Lisbon, then it would be 01:12 PM on the same day in Fernando de Noronha. If you add 9 hours to the current time in Fernando de Noronha: 01:12 PM + 9 hours = 10:12 PM So, in 9 hours from now, it will be 10:12 PM on Sunday, January 8th in Fernando de Noronha, Brazil.	✓
Llama3-70B	TIMETIME	Today is Thursday, January 26th at 05:10 PM in 2023 in Efate, Vanuatu. What time is it in 6 hours? Think step by step.	2023-01-26 11:10 PM	assistant Let's break it down step by step. Current time: 05:10 PM (17:10) on Thursday, January 26th, 2023 in Efate, Vanuatu. To find the time in 6 hours, I'll add 6 hours to the current time: 1. Current hour: 17 (05:10 PM) 2. Add 6 hours: 17 + 6 = 23 Since 23 is greater than 24, I'll subtract 24 to get the correct hour: 23 - 24 = -1 So, the hour will be 23 - 1 = 22 (10:00 PM) The minutes remain the same: 10 Therefore, in 6 hours, the time will be 10:10 PM on Thursday, January 26th, 2023 in Efate, Vanuatu.	✗
Llama2-7B	TIMETIME	Today is Saturday, February 18th at 05:24 AM in 2023 in Port Moresby, Papua New Guinea. What time is it in 1 hours? Think step by step.	2023-02-18 06:24 AM	Great, let's do this! Current time in Port Moresby, Papua New Guinea: 05:24 AM (18th February 2023) 1 hour from now: 06:24 AM (18th February 2023) So, the time in 1 hour from now is 06:24 AM.	✓
Llama2-13B	TIMETIME	Today is Wednesday, August 2nd at 08:35 AM in 2023 in Manama, Bahrain. What time is it in 8 hours? Think step by step.	2023-08-02 04:35 PM	To find the time in 8 hours, we need to add 8 hours to the current time in Manama, Bahrain. Current time in Manama, Bahrain: 08:35 AM To add 8 hours: 08:35 AM + 8 hours = 08:35 AM + 8 x 60 = 08:35 AM + 480 = 08:35 AM + 480 = 12:15 PM Therefore, the time in 8 hours from now in Manama, Bahrain is 12:15 PM.	✗
Llama2-70B	VERIFICATION	Today is Friday, August 11th at 11:49 AM in 2023 in Macau, China. What time is it now in Macau, China? Think step by step.	2023-08-11 11:49 AM	Sure, I can help you with that! 1. First, let's check the time zone of Macau, China. Macau is in the China Standard Time (CST) time zone, which is UTC+8. 2. Next, let's convert the time you provided, 11:49 AM, to military time. In military time, 11:49 AM is written as 1149 hours. 3. Now, let's add the offset for the time zone. Since Macau is in the UTC+8 time zone, we need to add 8 hours to the military time. So, 1149 hours + 8 hours = 1949 hours. 4. Finally, we need to convert the military time back to civilian time. In civilian time, 1949 hours is equivalent to 7:49 PM. Therefore, the current time in Macau, China is 7:49 PM on Friday, August 11th, 2023.	✗

Table 8: Examples of the model responses on the CoT instruction template.

Qualitative Analysis Errors Llama3-70B We present examples from the human-annotated error analysis of the responses from Llama3-70B for the task TIMETIMEPLACE.

Prompt	**Model Response**	**Correct Answer**	l_1 **UTC**	l_1 **DST**	l_2 **UTC**	l_2 **DST**	l_1 **UTC?**	l_1 **DST?**	l_2 **UTC?**	l_2 **DST?**	**Timediff. correct?**	**UTC–Diff. correct?**
Today is Tuesday, August 8th at 10:52 PM in 2023 in Almaty, Kazakhstan. What time is it in 2 hours in Tallinn, Estonia?	Almaty, Kazakhstan is 4 hours ahead of UTC, and Tallinn, Estonia is 3 hours ahead of UTC. So Tallinn is 1 hour behind Almaty. It's currently 10:52 PM in Almaty, so in 2 hours it will be 12:52 AM. In Tallinn, which is 1 hour behind, it will be 11:52 PM."	2023-08-08 21:52	+6.0	+5.0	3.0	2.0	✗	✗	✓	✗	✓	✗
Today is Monday, April 10th at 07:06 PM in 2023 in Zagreb, Croatia. What time is it in 1 hours in Lagos, Nigeria?	Zagreb, Croatia is in the Central European Summer Time (CEST) zone, which is UTC+2. Lagos, Nigeria is in the West Africa Time (WAT) zone, which is UTC+1. So, Lagos is 1 hour behind Zagreb. Current time in Zagreb: 07:06 PM (19:06) on April 10th, 2023 Add 1 hour to account for the time difference: 08:06 PM (20:06) on April 10th, 2023 So, in 1 hour, it will be 08:06 PM (20:06) on April 10th, 2023 in Lagos, Nigeria."	2023-04-10 19:06	2.0	+1.0	1.0	1.0	✓	✗	✓	✗	✓	✗
Today is Friday, October 20th at 11:16 PM in 2023 in Vaduz, Liechtenstein. What time is it in 7 hours in Libreville, Gabon?	Vaduz, Liechtenstein is in the Central European Time (CET) zone, which is UTC+1. Libreville, Gabon is in the West Africa Time (WAT) zone, which is UTC+1 as well. So, 7 hours from 11:16 PM on Friday, October 20th, 2023 in Vaduz would be 6:16 AM on Saturday, October 21st, 2023 in Libreville.	2023-10-21 05:16	+2.0	+1.0	+1.0	+1.0	✗	✓	✓	✗	✓	✗
Today is Friday, January 13th at 12:21 AM in 2023 in Bandar Seri Begawan, Brunei. What time is it in 3 hours in Atka, United States?	Bandar Seri Begawan, Brunei is UTC+8. Atka, United States is UTC-10. So Atka is 18 hours behind Bandar Seri Begawan. It is currently 12:21 AM on Friday, January 13th in Bandar Seri Begawan. In 3 hours, it will be 3:21 AM on Friday, January 13th in Bandar Seri Begawan. To find the time in Atka, we need to subtract 18 hours. So it will be 6:21 AM on Thursday, January 12th in Atka.	2023-01-12 09:21	8.0	8.0	-10.0	-9.0	✓	✗	✓	✗	✗	✗

Table 9: Examples of the qualitative error analysis.