Is the Future Predictable? Modeling Future Predictability for Reliable Temporal Reasoning

Anonymous ACL submission

Abstract

Large language models (LLMs) remain unreliable when reasoning about the future. Existing models often overlook the difference among future states, committing to specific answers even when the queried facts are inherently volatile. To capture this challenge, we introduce **ChronoRegime**, a future-oriented temporal reasoning task that requires models to determine whether historical temporal evidence justifies commitment to a future answer. Furthermore, we propose **C**onfidence-Aware **T**emporal **R**easoning (**CTR**), a framework that estimates future predictability from temporal dynamics and regulates answer commitment accordingly. Rather than performing unconditional forecasting, CTR commits only when the reconstructed evidence supports the target future state. Across multiple LLMs, CTR improves strict overall accuracy by 6.7 percentage points and reduces hallucination rates by 18.4 percentage points on average over the plain QA baseline. These results demonstrate that explicit predictability assessment is an effective mechanism for safer and more reliable temporal reasoning under future uncertainty.

1 Introduction

Large language models (LLMs) have demonstrated impressive capabilities in factual reasoning and question answering (Fatemi et al., 2024; Wang et al., 2024; Hu et al., 2025). However, their reliability on time-sensitive queries remains a fundamental and largely unresolved challenge within temporal reasoning (Dhingra et al., 2022; Wei et al., 2023). Although advanced knowledge-augmentation strategies mitigate factual inconsistencies by grounding outputs in external knowledge (Zhao et al., 2024; Bajpai et al., 2024), these systems remain highly susceptible to overconfident yet erroneous predictions when historical evidence is insufficient or the queried event is inherently unpredictable (Wen et al., 2025; Zhou et al., 2026).

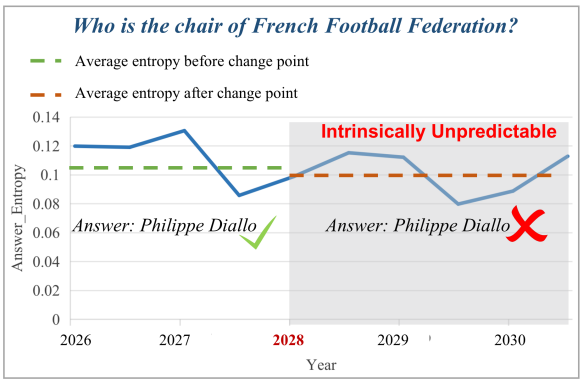

Figure 1: An example of how LLaMA3.1 performs in future-oriented temporal reasoning tasks. The model is unable to distinguish the predictable boundaries, and gives confident but incorrect answers at times that are inherently unpredictable.

This reliability gap becomes especially acute in future-oriented temporal reasoning, where the predictability of the target fact varies across queries, time horizons, and temporal evolution patterns (Vu et al., 2024; Wu et al., 2024; Zeng et al., 2025). For instance, as shown in Figure 1, historical evidence may justify predicting *"Philippe Diallo"* as the chair of the French Football Federation before 2028, but after the change point, the same historical evidence does not determine who will occupy the position in the future. More broadly, future queries span different evolution regimes, as illustrated in Figure 2: an unchanging fact such as the *"capital of France"* can be predicted reliably; whereas a fast-changing query such as *"Where will EMNLP be held as of 2028?"* may remain intrinsically unpredictable even with complete historical signals.

However, our preliminary study in Section 3 shows that current LLMs still struggle in future-oriented temporal reasoning, often committing to a specific future answer even when the underlying temporal evidence does not justify such commitment (e.g., predicting *"London, UK"* for the lo-

cation of EMNLP in 2028). This persistent drop, together with uniformly weak boundary-aligned commitment behavior across models, highlights a critical gap in time-sensitive forecasting: the inability to assess whether available evidence justifies a factual commitment.

Motivated by these findings, we identify a distinct and more challenging task in future-oriented temporal reasoning: **ChronoRegime**. We define ChronoRegime as the ability to infer the predictability regime of a future fact from historical temporal signals and to calibrate prediction commitment accordingly. Rather than merely producing a plausible future answer, the task asks whether the queried future state is predictable enough to warrant prediction.

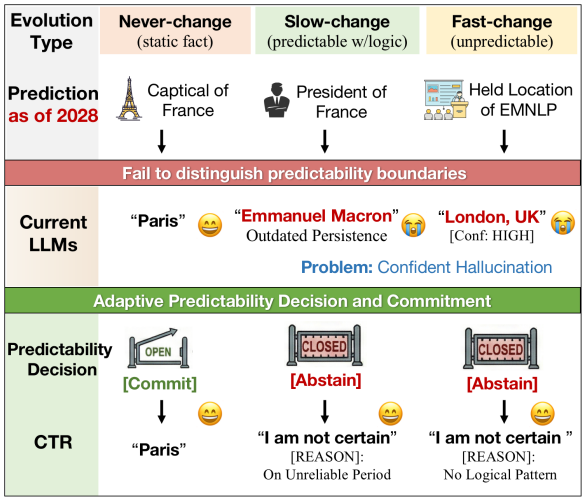

Figure 2: Challenge of Current LLMs on future predictability modeling and the motivation of ChronoRegime in highlighting the needs for decision-making beyond unconditional prediction.

We further propose **Confidence-Aware Temporal Reasoning (CTR)**, a framework for predictability-aware future factual reasoning. Rather than directly committing to an answer, CTR first estimates whether the queried fact lies in a predictable regime by modeling temporal evolution patterns in historical evidence, and only then decides whether to answer or abstain. This design shifts the task from unconditional forecasting to evidence-bounded commitment.

We evaluate CTR on updated versions of FRESHQA (Vu et al., 2024) and TEMPLAMA (Dhingra et al., 2022), using multiple LLM backbones, including LLaMA-3.1, Qwen3, and GPT-4o. CTR consistently improves future factual reasoning across models and substantially reduces overconfident failures on volatile queries, lowering hallucination rates by up to 18.4 percentage points on average. These results suggest that explicit predictability assessment provides an effective mechanism for calibrating model commitment under temporal uncertainty.

In summary, our contributions are as follows:

- We identify future predictability modeling as a central challenge in future-oriented temporal reasoning and introduce **ChronoRegime**, a task that evaluates whether historical temporal evidence justifies commitment to a future prediction.
- We empirically show that current LLMs fail on future-oriented questions not merely due to missing or outdated knowledge, but because they cannot distinguish predictable from volatile future facts.
- We propose **CTR**, a predictability-aware framework that models temporal evolution signals and regulates answer commitment for more reliable temporal reasoning under future uncertainty.

2 Related Work

Temporal reasoning has long been a central topic in the study of large language models (Vashishtha et al., 2020; Hu et al., 2023; Gurnee and Tegmark, 2023; Beniwal et al., 2024). A substantial body of work has evaluated LLMs' temporal reasoning capabilities from multiple perspectives, leading to the development of benchmark datasets and evaluation metrics that assess factual correctness (Fatemi et al., 2024; Chu et al., 2024; Zhu et al., 2025; Kasai et al., 2023), consistency (Bajpai et al., 2024; Jia et al., 2024), and reasoning traceability (Xu et al., 2023; Bazaga et al., 2025; Zhu et al., 2025). These studies consistently show that temporal factuality in LLMs is fundamentally challenged by the static and temporally mixed nature of pretraining corpora, which often results in outdated predictions, averaging of conflicting facts across time, and poorly calibrated confidence when models are queried about future states (Dhingra et al., 2022; Vu et al., 2024; Bajpai et al., 2024; Kasai et al., 2023).

One line of research improves LLMs' temporal awareness through pretraining or fine-tuning strategies that incorporate explicit temporal signals, with the goal of enhancing temporal scope understanding and confidence calibration (Ning et al., 2018;

Dhingra et al., 2022; Yang et al., 2023; Xiong et al., 2024a). These approaches typically treat time as a conditioning variable that constrains the validity of factual knowledge. Another line of work enhances temporal reasoning through structured representations, most notably knowledge graph–based approaches that explicitly encode entities, relations, and temporal constraints (Vu et al., 2024; Xiong et al., 2024b; Wang and Zhao, 2024; Hu et al., 2025). But they typically rely on predefined schemas, which makes graph updates costly and complex in dynamic settings where facts continuously evolve, thereby limiting their practicality for future-oriented reasoning that requires timely knowledge updates.

Another prominent direction augments LLMs with external retrieval mechanisms to refresh factual knowledge and reduce staleness (Vu et al., 2024; Wu et al., 2024). Retrieval-augmented and deep search methods leverage web search, document retrieval, or similar tools to improve performance on time-sensitive queries(Xu et al., 2023; Fu et al., 2024). However, most RAG-style pipelines implicitly treat retrieved documents as reliable evidence, despite differences in publication time and potential conflicts across sources. The lack of explicit modeling of temporal validity and uncertainty allows errors to propagate through reasoning chains, resulting in hallucinations and inconsistencies (Trivedi et al., 2023; Vu et al., 2024; Singal et al., 2024; Bajpai et al., 2024).

Existing studies have substantially advanced temporal reasoning in LLMs, especially in temporal factuality evaluation, temporal adaptation, structured knowledge modeling, and retrieval-based knowledge refreshing. By comparison, future-oriented temporal reasoning remains underexplored: how a model should reason about future states from historical temporal evidence under uncertainty is still not well understood.

3 Preliminary Study

We first evaluate current LLMs' ability to predict future facts from historical evidence through a controlled boundary-aligned probing study.

The setup involves a factual query on a future time, whose answer varies with respect to a certain time point, referred to as *changepoint*. Prior to the change point, the answer is definite and well supported with sufficient evidence from existing knowledge; after the change point, the same source of knowledge becomes insufficient for determining the same answer. For example, as shown in Figure 3, one is asked to predict the location of most recent Summer Olympics. As of year 2028, one can deduce the correct answer with high confidence based on the information of the current Summer Olympics and the period of the event. After 2028, however, the answer to the same question becomes inherently indeterminate without additional information on the "next Summer Olympics".

This study evaluates whether LLM outputs exhibit appropriate commitment behavior when target times move from an evidence-supported region to a region where the same evidence no longer justifies a concrete prediction. A reliable model should provide confident answers when evidence is sufficient, but revise, abstain, or express greater uncertainty once the evidence no longer supports a prediction.

3.1 Problem Setting

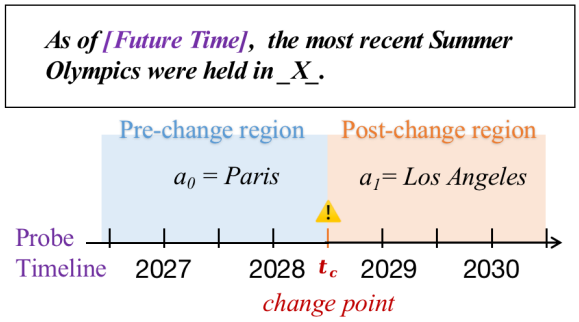

Figure 3: Illustration of the boundary-probing setup. The change point acts as a predictability boundary: before it, the anchor answer is supported by temporal evidence; after it, the model must revise its commitment or avoid unsupported prediction.

For evaluation, each instance consists of a factual query, an anchor answer a_0 at the present time as a representation of the existing knowledge, a change pointt_c as the transition boundary, and a post-boundary answer a_1 when available.

We probe the model at a series of time stamps along the "future time line", spanning both pre-change and post-change regions. At each probe, the model is prompted with the time-conditioned query for the target timestamp. The annotated change point t_c and the post-boundary answer a_1 are hidden from the model and used only for evaluation.

In our experiments, probes are placed at half-year offsets within a three-year window around the change point, and the boundary-aligned probe is counted as post-change. The probing instances

are drawn from the curated temporal QA pool described in Section 5.1.

3.2 Evaluation Metrics

We evaluate boundary-aligned temporal reasoning with a compact set of metrics, and provide their formal definitions in Appendix B.

Pre- and Post-change Accuracy (Pre/Post Acc) measure answer correctness before and after the change point, respectively.

Change-aware Success(CAS) evaluates whether the model correctly switches its commitment across the nearest change probes.

Post-change Persistence Rate (Post-PR) measures outdated persistence,i.e., whether the model continues to repeat the expired anchor answer after the boundary.

Wrong Concrete Commitment Rate (WCCR) measures unsafe post-change behavior by counting concrete non-abstention predictions that are wrong.

Entropy Shift(ΔH) denotes the post-minus-pre shift in answer entropy, capturing whether model uncertainty increases after the anchor answer loses validity.

3.3 Analysis

Model	PreAcc ↑	PostAcc ↑	CAS ↑	Post-PR ↓	WCCR ↓	ΔH
LLaMA	0.500	0.158	0.000	0.429	0.832	0.006
Qwen3	**0.911**	0.219	0.250	0.924	0.786	0.002
Mistral	0.333	0.082	0.036	**0.276**	0.867	0.001
GPT-4o	0.839	**0.316**	**0.286**	0.689	**0.643**	**0.022**

Table 1: Boundary-probing results.

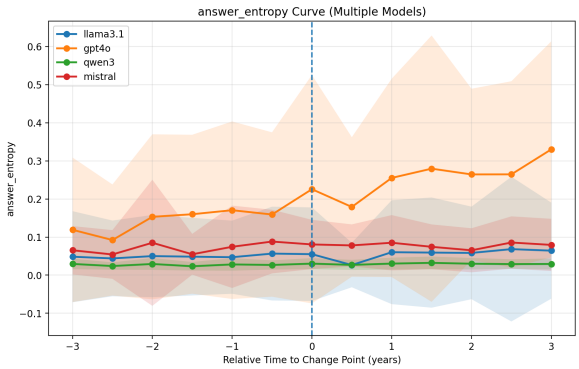

Figure 4: Average answer entropy around factual change points. Across models, uncertainty remains largely flat before and after temporal transitions, indicating that current LLMs do not reliably increase uncertainty when future commitment becomes less justified.

Table 1 shows that current LLMs can often produce the anchor fact for pre-boundary target times, but fail to adjust their commitment once the queried target time falls beyond the annotated support boundary. Qwen3 and GPT-4o achieve high PreAcc scores of 0.911 and 0.839, respectively, yet their PostAcc drops to 0.219 and 0.316. CAS is also low across models, with the best result only 0.286 from GPT-4o, indicating weak boundary-aware switching.

Post-change failures are mostly unsafe commitments rather than cautious abstentions. WCCR remains high across models, ranging from 0.643 to 0.867. Qwen3 and GPT-4o further show strong outdated persistence, repeating the expired anchor answer in 92.4% and 68.9% of post-change probes. Mistral has lower persistence but the highest WCCR, suggesting unsupported alternative guesses. ΔH remains close to zero, showing that uncertainty barely rises after the boundary.

Figure 4 further shows that model uncertainty remains largely flat around the annotated transition boundary. The entropy shift ΔH is close to zero across all models, suggesting that they do not become meaningfully less confident when the anchor answer loses validity. This result indicates that current LLMs fail to infer the loss of evidential support from the time-conditioned query alone.

These findings motivate the central design of ChronoRegime and CTR. Reliable future-oriented reasoning requires more than predicting a plausible future answer; it requires estimating whether the available temporal evidence is sufficient to justify commitment.

4 Methodology

We introduce **Confidence-Aware Temporal Reasoning (CTR)**, a framework for predictability-aware future factual reasoning under the **ChronoRegime** formulation. CTR follows a simple principle: before committing to a future answer, the model should first estimate whether the answer is predictable from temporal evidence. The overall architecture is illustrated in Figure 5.

4.1 ChronoRegime Formulation

Given a factual query q and a future time t^*, ChronoRegime asks whether the future factual state a_{t^*} is predictable from bounded historical temporal evidence. We define predictability in an evidence-bounded sense: a future state is pre-

dictable if the available temporal evidence supports a concrete answer with sufficient reliability; otherwise, the model should avoid commitment. To operationalize this decision, we distinguish between two levels of temporal reasoning. First, the model infers a factual evolution regime $z \in \{\text{FAST}, \text{SLOW}, \text{NEVER}\}$, where z describes the volatility profile of the queried fact. Second, the model maps this evolution regime, together with the reconstructed historical timeline and target time t^*, into a predictability decision $u \in \{\text{PREDICTABLE}, \text{UNPREDICTABLE}\}$. The final output is therefore $\langle r, z, u, \hat{a}_{t^*} \rangle$, where r is a reasoning trace explaining the decision and $\hat{a}_{t^*}$ is the predicted fill-in value for q as of year t^*.

ChronoRegime treats predictability as an evidence-bounded decision u: the model should commit only when the inferred regime, historical evidence, and target time jointly support a concrete answer. The benchmark-specific mapping from evolution regimes to predictability labels is described in Section 5.1

4.2 CTR Architecture

CTR deliberately adopts conservative heuristics rather than optimized predictors. Its goal is not to maximize change-detection recall, but to regulate future commitment under uncertainty and avoid unfounded speculation when temporal evidence is weak or inconsistent.

Driven by this principle, CTR is formulated as a confidence-driven decision pipeline that progressively transforms noisy historical signals into a structured predictability regime for downstream commitment control. The framework consists of five-stage process: (1) Historical Recall; (2) Timeline Reconstruction; (3) Change Point Detection; (4) Predictability Regime Induction and (5) Commitment-Regulated Inference.

Historical Recall Given a future time t^*, CTR performs fine-grained historical probing over a window of past years:

$$\mathcal{Y} = \{t^* - W, \ldots, t^* - 1\}. \tag{1}$$

For each target year $y \in \mathcal{Y}$, the model is queried to produce a predicted fill-in value a_y along with its token-level uncertainty, measured by the average generation entropy H_y.

To obtain a calibrated reliability signal, we transform the raw entropy into a base confidence score c_y via a piecewise linear mapping $f(H_y)$. This transformation normalizes the entropy within a predefined uncertainty range $[H_{\min}, H_{\max}]$ and inverts it:

$$c_y = \begin{cases} 1, & H_y < H_{\min} \\ 1 - \frac{H_y - H_{\min}}{H_{\max} - H_{\min}}, & H_{\min} \le H_y \le H_{\max} \\ 0, & H_y > H_{\max} \end{cases} \tag{2}$$

where we empirically set $H_{\min} = 0$ and $H_{\max} = 0.05$. The resulting sequence of historical recall probes is represented as:

$$\mathcal{R} = \{(y, a_y, c_y)\}_{y \in \mathcal{Y}} \tag{3}$$

This sequence $\mathcal{R}$ serves as the foundational evidence for subsequent timeline reconstruction and change point detection, where c_y acts as the intrinsic weight for each temporal observation.

Timeline Reconstruction The recall probes are organized into a temporal timeline

$$\mathcal{T} = \{(y, v_y, q_y)\}, \tag{4}$$

where $v_y = a_y$ denotes the recalled factual state at time y, and q_y is a corrected reliability weight.

Importantly, q_y is not directly inherited from the entropy-based confidence c_y. Instead, CTR applies a probe-quality correction function that down-weights unreliable probes, such as those exhibiting abnormally long outputs, repetitive generation patterns, or malformed responses. Formally, we define

$$q_y = \text{CLIP}_{[0,1]}\big(c_y \cdot (1 - \rho_{\text{len}}(y)) \cdot (1 - \rho_{\text{rep}}(y))\big), \tag{5}$$

where ρ_{len} and ρ_{rep} denote mild penalties based on output length and n-gram repetition, respectively. This correction reflects the fact that historical recall itself may be noisy and should not be treated as uniformly reliable.

Change Point Detection Given the reconstructed timeline $\mathcal{T}$, CTR aims to identify years in which the factual state undergoes a genuine transition. We adopt a procedure that combines candidate proposal with consistency-aware reweighting.

Candidate proposal: We first obtain a set of candidate change-point years $\tilde{\mathcal{C}}$ by prompting the model to inspect the timeline and identify major state transitions. If the model output is unavailable or invalid, we fall back to a deterministic rule that proposes time y whenever $v_y \neq v_{y-1}$. All candidates are restricted to years explicitly appearing in the timeline.

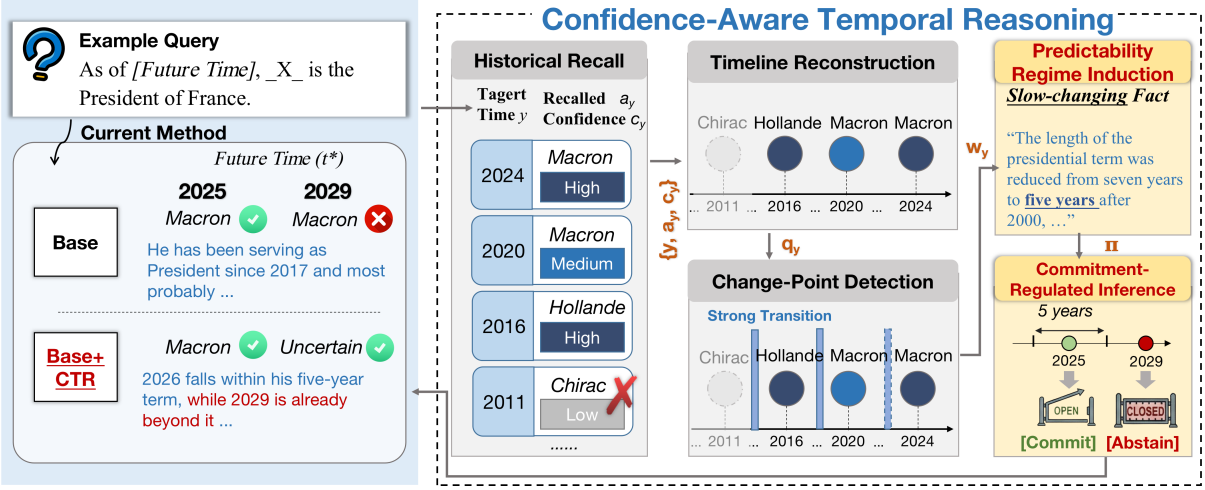

Figure 5: **Overview for Confidence-Aware Temporal Reasoning (CTR).** CTR leverages entropy-based confidence to shift from static retrieval to dynamic evolution modeling. Notably, its confidence-aware mechanism suppresses erroneous historical evidence (e.g., the low-confidence 2011 entry) to mitigate error propagation. By inducing explicit evolution patterns, CTR overcomes the persistence bias seen in the traditional reasoning modes and prioritizes reliability, avoiding hallucinated extrapolation for volatile future queries.

Consistency-aware reweighting: Each candidate $y \in \tilde{\mathcal{C}}$ is then filtered and assigned a confidence weight. Candidates with no effective state change ($v_y = v_{y-1}$) are discarded. For the remaining candidates, we compute

$$w_y = \sqrt{q_{y-1}\, q_y} \cdot g_y, \tag{6}$$

where w_y denotes the confidence of the change point at time y and $g_y \in (0, 1]$ penalizes isolated single-year deviations.

Specifically, a candidate is treated as singleton noise if it is unsupported by its immediate temporal context, i.e., $v_{y-1} = v_{y+1}$ but $v_y \neq v_{y-1}$, and its probe reliability is substantially lower than that of neighboring years.

In such cases, g_y is reduced whenever

$$q_y < \tau_{\text{single}} \cdot \min(q_{y-1}, q_{y+1}), \tag{7}$$

thereby enforcing a local temporal consistency constraint. This design favors transitions that persist across time and suppresses spurious one-off fluctuations.

The resulting set of weighted change points is denoted as

$$\mathcal{C} = \{(y, w_y)\}. \tag{8}$$

Predictability Regime Induction Given the reconstructed timeline $\mathcal{T}$ and weighted change points $\mathcal{C}$, CTR induces a high-level temporal evolution pattern by aggregating volatility statistics over the timeline. Based on the frequency and persistence of state changes, the evolution regime z is inferred. Then, CTR maps z, together with the target time t^* and the reconstructed support in $\mathcal{T}$, into a predictability decision u. The induced pattern representation is therefore

$$\pi = \langle z, u, \text{key_evidence} \rangle, \tag{9}$$

which serves as an explicit evolution hypothesis for subsequent inference.

Commitment-Regulated Inference CTR performs future inference conditioned on the induced pattern representation π, where the inferred evolution type serves as the ChronoRegime decision signal for regulating downstream commitment. In this way, CTR first determines whether the future fact is predictable under the inferred temporal regime, and only then decides what answer to output.

The model predicts the factual state at the future time t^* as

$$\hat{a}_{t^*} = \arg\max_{a} p(a \mid q, t^*, \pi), \tag{10}$$

subject to strict constraints that prohibit unsupported extrapolation.

5 Experiments

5.1 Dataset

We construct a diagnostic dataset from temporal factual QA benchmarks to evaluate temporal evolution and confidence-aware commitment.

Model	Variant	Answer Accuracy (%) ↑				Type Acc. (%) ↑	Halluc. (%) ↓
		Fast	Slow	Never	Overall		
LLaMA3.1	Base	**6.9**	21.0	50.0	26.7	23.7	36.8
	+Controlled Retrieval	5.9	29.4	53.4	30.6	26.1	33.2
	+CTR	**6.9**	**31.9**	**56.0**	**32.6**	**55.3**	**22.6**
Qwen3	Base	4.9	26.9	41.4	25.2	35.0	56.7
	+Controlled Retrieval	**8.8**	21.0	47.4	26.4	54.9	54.6
	+CTR	6.9	**32.8**	**57.8**	**33.5**	**57.5**	**25.5**
Mistral	Base	7.8	12.6	50.0	24.0	29.7	40.1
	+Controlled Retrieval	4.9	16.0	42.2	21.7	51.9	36.8
	+CTR	**8.8**	**21.0**	**50.9**	**27.6**	**52.8**	**18.4**
GPT-4o	Base	16.7	40.3	74.1	44.8	48.1	17.8
	+Controlled Retrieval	13.7	46.2	79.3	47.8	**63.8**	13.7
	+CTR	**18.6**	**52.9**	**85.3**	**53.7**	**63.8**	**11.3**

Table 2: Main results (%) on answer accuracy, type accuracy, and hallucination rate. Controlled Retrieval reasoning yields inconsistent gains over plain QA, while CTR consistently improves performance on slow- and never-changing facts and reduces hallucinations across models.

Dataset Construction We sample from FRESHQA [1] and TEMPLAMA [2] rather than use them in full, since ChronoRegime requires clear evidence-bounded predictability labels. An instance is labeled PREDICTABLE only if a unique future answer is supported by stable persistence, a known boundary, or a deterministic rule/cycle; it is labeled UNPREDICTABLE if the target value depends on external updates or future events outside the bounded evidence. We refresh the gold annotations to reflect the world state as of *September 2025*. This refresh is used only to correct stale answers in the source benchmarks and to define evaluation targets. We also annotate each instance with an evolution regime, a predictability label, and optional *next_answer*. Details are in Appendix G.

Dataset Statistics The final dataset contains 337 base factual queries: 102 *fast-changing*, 119 *slow-changing*, and 116 *never-changing* facts. Each instance can be instantiated at multiple target times. Dataset details and additional examples are provided in the Appendix H.

5.2 Baselines

Models We evaluate CTR on both open-source and closed-source LLMs, including **LLaMA-3.1-8B-Instruct** (Dubey et al., 2024), **Qwen3-4B-Instruct** (Yang et al., 2025), **Mistral-7B-Instruct-v0.3** (Jiang et al., 2023), and **GPT-4o**. To avoid temporal leakage, we use model snapshots whose documented knowledge cutoff precedes 2025, while all target queries satisfy $t^* \geq 2025$. All models are evaluated in a zero-shot setting with identical prompts and decoding configurations.

Controlled Retrieval Baseline We compare CTR with a controlled retrieve-then-reason baseline based on Chain-of-Thought (Wei et al., 2022) reasoning. The baseline uses the same bounded historical evidence as CTR, allowing us to isolate the effect of CTR's confidence-aware decision mechanism rather than access to historical information alone. Details on model checkpoints and the rationale for using bounded evidence instead of open-corpus RAG are provided in Appendix D.

5.3 Experimental Setup

All experiments are conducted with a fixed future time $t^* \geq 2025$. Unless otherwise specified, CTR probes a historical window of size $W = 50$, querying years $\{t^*-50, \ldots, t^*-1\}$. All models are evaluated in a zero-shot setting with a decoding temperature of 0.2 and *max_new_tokens* set to 1024. Results are averaged over five runs.

Hallucination is measured using a two-stage verification protocol, with details provided in Appendix F.

6 Result Analysis

6.1 Overall Performance

Table 2 compares the Base models, the retrieval-guided baseline, and the proposed CTR framework. Overall, CTR consistently improves answer accuracy and evolution type prediction while substan-

[1] https://github.com/freshllms/freshqa

[2] https://github.com/google-research/language/tree/master/language/templama

tially reducing hallucination, demonstrating more reliable future reasoning.

CTR outperforms both baselines in answer accuracy across all backbone models, with the largest gains on Slow- and Never-changing facts. On GPT-4o, accuracy on Slow-changing facts increases from 46.2% to 52.9% and on Never-changing facts from 79.3% to 85.3% compared to prompting. Importantly, these gains do not degrade performance on Fast-changing facts. GPT-4o with CTR achieves 18.6% accuracy on Fast facts, exceeding both the Base (16.7%) and Controlled Retrieval (13.7%) settings, indicating that CTR selectively commits to predictions rather than suppressing them under uncertainty.

6.2 Comparison with Controlled Retrieval Reasoning

Controlled Retrieval reasoning yields limited and inconsistent improvements over the Base configuration. While it enhances overall accuracy for specific models, such as lifting LLaMA3.1 from 26.7% to 30.6%. These gains fail to generalize across all temporal fact types. Crucially, Controlled Retrieval can degrade performance on fast-changing facts, as seen in Mistral's Fast-fact accuracy drop from 7.8% to 4.9%. In sharp contrast, CTR leverages explicit confidence calibration to consistently secure the best performance across all architectures, elevating Qwen3's Type Accuracy to a peak of 57.5%.

6.3 Reliability: Hallucination and Abstention

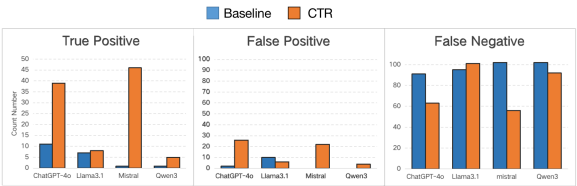

Figure 6: Abstention performance of CTR policy.

CTR also yields the largest improvements in reliability. Hallucination rates are consistently lower than both baselines across all models. For Qwen3, hallucination decreases from 56.7% in the Base setting to 25.5% under CTR, far below the controlled retrieval result of 54.6%. GPT-4o exhibits a similarly low hallucination rate of 11.3%.

This reliability gain is further reflected in abstention behavior. We treat explicit non-commitment outputs as abstentions. Briefly, TP denotes abstaining on unpredictable instances, FP denotes abstaining on predictable instances, and FN denotes making a concrete prediction on unpredictable instances; formal counting rules are provided in Appendix B.4.

As shown in Figure 6, CTR generally reduces False Negatives on unpredictable cases,for example, from 91 to 63 on GPT-4o and from 102 to 56 on Mistral, while also increasing True Positives. Although False Positives increase for some backbones, this suggests a shift toward a more conservative commitment policy rather than unconditional extrapolation.

Above results support the central ChronoRegime hypothesis: reliable future reasoning depends not only on predicting correctly when evidence is sufficient, but also on refraining from unsupported commitments when predictability is weak.

6.4 Case Study

We further analyze representative cases to examine how CTR repairs unreliable future reasoning. The case studies show that controlled retrieval reasoning often commits to plausible but weakly supported answers, including unsupported extrapolations, fabricated facts, and noisy change-point interpretations. CTR mitigates these failures by aggregating temporal evidence, suppressing isolated low-confidence changes, and committing only when long-term stability supports the prediction. Full case analyses are provided in Appendix E.

7 Conclusion

In this work, we study future-oriented temporal reasoning through the lens of prediction commitment. We introduce **ChronoRegime**, a task that requires models to determine whether historical temporal evidence is sufficient to justify a future prediction. Our empirical analysis shows that current LLMs often struggle with this judgment, leading to overconfident commitments under temporal uncertainty.

To address this issue, we propose **Confidence-Aware Temporal Reasoning (CTR)**, which separates predictability estimation from answer commitment. Across four LLMs, CTR consistently improves answer accuracy, reduces hallucination, and provides more transparent grounding for future predictions. These findings highlight the importance of modeling predictability, rather than merely extrapolating future answers, for reliable temporal reasoning.

Limitations

While CTR demonstrates consistent improvements in future-oriented factual reasoning, several limitations remain.

Computational efficiency. Our primary experimental setup adopts a rigorous year-by-year probing strategy ($W = 50$) to establish a diagnostic upper bound for temporal reasoning performance. We acknowledge that this incurs a linear computational cost ($\mathcal{O}(W)$) relative to the window size. However, the historical recall process is fully parallelizable. Furthermore, for real-world deployment, CTR is designed to function primarily as an offline knowledge curation pipeline, constructing reliable temporal indices once to support low-latency lookups for subsequent queries. Future iterations can further optimize efficiency by implementing adaptive sampling (e.g., binary search) to reduce the probing complexity from linear $\mathcal{O}(W)$ to logarithmic $\mathcal{O}(\log W)$.

Uncertainty estimation. CTR uses token-level entropy as a proxy for model confidence. While this signal is model-agnostic and easy to compute, it may not fully capture epistemic uncertainty, particularly for models with imperfect probability calibration. Exploring alternative or complementary uncertainty measures remains an important direction for future work.

Scope of evaluation. Our evaluation focuses on cloze-style temporal factual question answering with discrete time points. While this setting enables controlled analysis of future commitment and abstention, it does not cover more complex scenarios such as open-ended forecasting, interacting events, or long-horizon planning, which remain open challenges for future exploration.

Ethics Statement

All work in this paper adheres to the ACL Code of Ethics. For the human evaluation of hallucinations, we recruited volunteer annotators and compensated them in accordance with local wage standards. The collected data do not involve any personally identifiable or sensitive information.

The study is conducted in a controlled research setting using released large language models under their respective licenses and usage policies. The proposed framework is intended for methodological analysis of future-oriented factual reasoning rather than real-world deployment. While model-generated outputs may contain inaccuracies, such risks are inherent to large language models and are not the focus of this work.

References

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.

Adrián Bazaga, Rexhina Blloshmi, Bill Byrne, and Adrià de Gispert. 2025. Learning to reason over time: Timeline self-reflection for improved temporal reasoning in language models. *arXiv preprint arXiv:2504.05258*.

Himanshu Beniwal, Dishant Patel, Kowsik Nandagopan, 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.

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)*, pages 1204–1228.

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.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, and 1 others. 2024. The llama 3 herd of models. *arXiv e-prints*, pages arXiv–2407.

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. *arXiv preprint arXiv:2406.09170*.

Chenhan Fu, Guoming Wang, Rongxing Lu, and Siliang Tang. 2024. Fastlearn: A rapid learning agent for chat models to acquire latest knowledge. *2024 IEEE 7th International Conference on Multimedia Information Processing and Retrieval (MIPR)*, pages 183–189.

Wes Gurnee and Max Tegmark. 2023. Language models represent space and time. *arXiv preprint arXiv:2310.02207*.

Nan Hu, Jiaoyan Chen, Yike Wu, Guilin Qi, Hongru Wang, Sheng Bi, Yongrui Chen, Tongtong Wu, and Jeff Z Pan. 2025. Can llms evaluate complex attribution in qa? automatic benchmarking using knowledge graphs. In *The 63rd Annual Meeting of the Association for Computational Linguistics (ACL 2025)*.

Xuming Hu, Junzhe Chen, Xiaochuan Li, Yufei Guo, Lijie Wen, Philip S Yu, and Zhijiang Guo. 2023. Do large language models know about facts? *arXiv preprint arXiv:2310.05177*.

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 Web Conference 2024*, pages 1394–1399.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

Jungo Kasai, Keisuke Sakaguchi, Ronan Le Bras, Akari Asai, Xinyan Yu, Dragomir Radev, Noah A Smith, Yejin Choi, Kentaro Inui, and 1 others. 2023. Realtime qa: What's the answer right now? *Advances in neural information processing systems*, 36:49025–49043.

Qiang Ning, Zhili Feng, Hao Wu, and Dan Roth. 2018. Joint reasoning for temporal and causal relations. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2278–2288.

Ronit Singal, Pransh Patwa, Parth Patwa, Aman Chadha, and Amitava Das. 2024. Evidence-backed fact checking using rag and few-shot in-context learning with llms. In *Proceedings of the Seventh Fact Extraction and VERification Workshop (FEVER)*, pages 91–98.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2023. Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions. In *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 1: long papers)*, pages 10014–10037.

Siddharth Vashishtha, Adam Poliak, Yash Kumar Lal, Benjamin Van Durme, and Aaron Steven White. 2020. Temporal reasoning in natural language inference. In *Findings*.

Tu Vu, Mohit Iyyer, Xuezhi Wang, Noah Constant, Jerry Wei, Jason Wei, Chris Tar, Yun-Hsuan Sung, Denny Zhou, Quoc Le, and 1 others. 2024. Freshllms: Refreshing large language models with search engine augmentation. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 13697–13720.

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.

Yuxia Wang, Minghan Wang, Muhammad Arslan Manzoor, Fei Liu, Georgi Nenkov Georgiev, Rocktim Jyoti Das, and Preslav Nakov. 2024. Factuality of large language models: A survey. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 19519–19529.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed H. Chi, F. Xia, Quoc Le, and Denny Zhou. 2022. Chain of thought prompting elicits reasoning in large language models. *ArXiv*, abs/2201.11903.

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.

Bingbing Wen, Jihan Yao, Shangbin Feng, Chenjun Xu, Yulia Tsvetkov, Bill Howe, and Lucy Lu Wang. 2025. Know your limits: A survey of abstention in large language models. *Transactions of the Association for Computational Linguistics*, 13:529–556.

Feifan Wu, Lingyuan Liu, Wentao He, Ziqi Liu, Zhiqiang Zhang, Haofen Wang, and Meng Wang. 2024. Time-sensitve retrieval-augmented generation for question answering. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pages 2544–2553.

Siheng Xiong, Ali Payani, Ramana Kompella, and Faramarz Fekri. 2024a. 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.

Siheng Xiong, Ali Payani, Ramana Rao Kompella, and Faramarz Fekri. 2024b. Large language models can learn temporal reasoning. *ArXiv*, abs/2401.06853.

Shicheng Xu, Liang Pang, Huawei Shen, Xueqi Cheng, and Tat seng Chua. 2023. Search-in-the-chain: Towards the accurate, credible and traceable content generation for complex knowledge-intensive tasks. *ArXiv*, abs/2304.14732.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Sen Yang, Xin Li, Lidong Bing, and Wai Lam. 2023. Once upon a time in graph: Relative-time pretraining for complex temporal reasoning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 11879–11895.

Zhiyuan Zeng, Jiashuo Liu, Siyuan Chen, Tianci He, Yali Liao, Yixiao Tian, Jinpeng Wang, Zaiyuan Wang, Yang Yang, Lingyue Yin, and 1 others. 2025. Futurex: An advanced live benchmark for llm agents in future prediction. *arXiv preprint arXiv:2508.11987*.

Bowen Zhao, Zander Brumbaugh, Yizhong Wang, Hannaneh Hajishirzi, and Noah A Smith. 2024. Set the clock: Temporal alignment of pretrained language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 15015–15040.

Xinyu Zhou, Chang Jin, Carsten Eickhoff, Zhijiang Guo, and Seyed Ali Bahrainian. 2026. When silence is golden: Can llms learn to abstain in temporal qa and beyond? *arXiv preprint arXiv:2602.04755*.

Zhiyuan Zhu, Yusheng Liao, Zhe Chen, Yuhao Wang, Yunfeng Guan, Yanfeng Wang, and Yu Wang. 2025. Evolvebench: A comprehensive benchmark for assessing temporal awareness in llms on evolving knowledge. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16173–16188.

A CTR Inference Algorithm

This appendix provides the detailed inference procedure of Confidence-Aware Temporal Reasoning (CTR). CTR is designed to separate two decisions that are often conflated in future-oriented temporal reasoning: (i) estimating the temporal predictability regime of the queried fact, and (ii) deciding whether the model should commit to a concrete future answer.

Algorithm 1 summarizes the full pipeline. Given a query q and a target future time t^*, CTR first probes historical states before t^*, reconstructs a temporal timeline, detects reliable change signals, induces a predictability regime, and finally applies a commitment policy. Unlike unconditional prediction, the final step may return an abstention expression when the inferred evidence does not justify commitment.

Algorithm 1 Confidence-Aware Temporal Reasoning (CTR)

Require: Query q, future time t^*, window size W
Ensure: Regime z and final output $\hat{a}$

Historical Recall
for $y = t^* - W$ to $t^* - 1$ **do**
  $(a_y, H_y) \leftarrow \mathrm{ProbeModel}(q, y)$
  $c_y \leftarrow f(H_y)$
end for
Temporal Evidence Construction
$\mathcal{T} \leftarrow \mathrm{BuildTimeline}(\{(y, a_y, c_y)\})$
$\mathcal{C} \leftarrow \mathrm{DetectChanges}(\mathcal{T})$
Predictability Estimation
$z, u \leftarrow \mathrm{InferRegime}(\mathcal{T}, \mathcal{C})$
$\pi \leftarrow \mathrm{SummarizePattern}(z, u, \mathcal{T}, \mathcal{C})$
Commitment Decision
$s \leftarrow \mathrm{SupportScore}(q, t^*, \pi)$
if $u = \mathrm{Unpredictable}$ **or** $s < \tau_{\mathrm{commit}}$ **then**
  $\hat{a} \leftarrow$ `Abstention`
else
  $\hat{a} \leftarrow \mathrm{Predict}(q, t^*, \pi)$
end if
return $(z, u, \hat{a})$

The commitment policy is intentionally conservative. For stable or slowly evolving facts, CTR commits only when the reconstructed timeline provides sufficient support for the future state. For highly volatile facts, or when the evidence contains inconsistent or low-confidence transitions, CTR abstains rather than extrapolating unsupported answers. Thus, the algorithm operationalizes ChronoRegime as an evidence-bounded decision process rather than an unconditional answer generator.

B Detailed Evaluation Metrics and Protocol

B.1 Boundary-Probing Metrics

Let K^- and K^+ denote the pre-change and post-change probe offsets, respectively, with the boundary-aligned probe included in K^+. For each instance, a_0 denotes the anchor answer, a_1 denotes the held-out post-change answer, and $\hat{a}_k$ denotes the model output at offset k. We use to denote empirical averaging over evaluated instances and probe offsets, and $\mathcal{A}_{\mathrm{abs}}$ for explicit abstention or uncertainty expressions.

Pre- and Post-change Accuracy We report accuracy separately before and after the change point:

$$\begin{aligned} \mathrm{PreAcc} &= _{k \in K^-} \, [\hat{a}_k = a_0], \\ \mathrm{PostAcc} &= _{k \in K^+} \, [\hat{a}_k = a_1]. \end{aligned} \tag{11}$$

PreAcc measures whether the model recovers the anchor answer when it is still valid, while PostAcc measures whether it adapts after the anchor answer expires.

Change-aware Success To directly assess whether a model captures the boundary transition, we define CAS as:

$$\mathrm{CAS} = [\hat{a}_{k^-} = a_0 \wedge \hat{a}_{k^+} = a_1], \tag{12}$$

where k^- and k^+ denote the nearest pre-change and post-change probes, respectively. CAS requires the model to be correct on both sides of the boundary, and therefore measures whether it can switch its commitment rather than being correct at isolated time points.

Post-change Persistence Rate We measure whether the model continues to repeat the anchor answer after it has become invalid:

$$\text{Post-PR} = _{k \in K^+} \, [\hat{a}_k = a_0]. \tag{13}$$

A high Post-PR indicates outdated persistence: the model treats the expired anchor answer as if it were still reliable after the predictability boundary.

Wrong Concrete Commitment Rate Post-change accuracy alone does not distinguish cautious uncertainty from concrete wrong prediction. We therefore define WCCR over post-change probes as:

$$\mathrm{WCCR} =_{k \in K^+} w_k, \quad w_k = [\hat{a}_k \notin \mathcal{A}_{\mathrm{abs}}] \cdot [\hat{a}_k \neq a_1]. \tag{14}$$

Here, $w_k = 1$ only when the model makes a concrete non-abstention prediction that does not match the post-change gold answer. Lower WCCR indicates safer behavior under weakened predictability support.

Entropy Shift We measure whether model uncertainty changes across the predictability boundary:

$$\Delta H =_{k \in K^+} H_k -_{k \in K^-} H_k. \tag{15}$$

A positive ΔH suggests that the model becomes more uncertain after the anchor answer loses validity. In contrast, $\Delta H \approx 0$ indicates that the model does not reflect the evaluator-annotated loss of prediction support.

B.2 Answer Correctness Criteria

Strict Correctness We primarily use strict correctness to evaluate whether the model identifies the exact factual state. This serves as our main accuracy criterion.

Lenient Post-change Correctness For future-oriented queries, strict evaluation may penalize reasonable caution. Therefore, we additionally introduce a lenient criterion that accepts explicit expressions of uncertainty, such as outputs containing "Unknown", "Uncertain", or "TBD", as valid responses for volatile facts:

$$\mathrm{Correct}_k^{\mathrm{lenient}} = \begin{cases} \mathrm{Correct}_k^{\mathrm{strict}}, & k \in K^-, \\ [\hat{a}_k \in \{a_1, \mathrm{Unknown}\}], & k \in K^+ \end{cases} \tag{16}$$

This metric distinguishes between correctable epistemic uncertainty and confident hallucinations.

B.3 Uncertainty Estimation

We quantify model confidence using token-level entropy derived from the decoding process. Let $\mathcal{T}_k$ denote the set of token indices corresponding to the generated answer $\hat{a}_k$. The average answer entropy for an instance at time offset k is defined as:

$$H_k = \frac{1}{|\mathcal{T}_k|} \sum_{j \in \mathcal{T}_k} \mathcal{H}\left(p_\theta(\cdot \mid x_{<j}, x_k)\right), \tag{17}$$

where p_θ is the model's predictive distribution and $\mathcal{H}(\cdot)$ denotes Shannon entropy. The entropy shift reported in Table 1 is computed according to Equation 15.

B.4 Abstention Evaluation

To evaluate whether a model knows when not to commit, we additionally report abstention-oriented statistics on unpredictable instances. Let $u_i = 1$ indicate that instance i is unpredictable and $u_i = 0$ indicate that it is predictable. We define TP, FP, and FN as:

$$\begin{aligned} \mathrm{TP} &= \sum_i [u_i = 1 \wedge \hat{a}_i \in \mathcal{A}_{\mathrm{abs}}], \\ \mathrm{FP} &= \sum_i [u_i = 0 \wedge \hat{a}_i \in \mathcal{A}_{\mathrm{abs}}], \\ \mathrm{FN} &= \sum_i [u_i = 1 \wedge \hat{a}_i \notin \mathcal{A}_{\mathrm{abs}}]. \end{aligned} \tag{18}$$

Thus, TP denotes abstaining on an unpredictable instance, FP denotes abstaining on a predictable instance, and FN denotes making a concrete prediction on an unpredictable instance.

C Prediction Flip Curves around Factual Change Points

Figure 7 plots model predictions around factual change points. For each relative time offset, we report the probability that the model predicts the pre-change anchor answer and the probability that it predicts the post-change answer. This figure complements the preliminary study by visualizing whether models actually switch from the old answer to the new answer after the temporal boundary.

Across models, the probability of predicting the anchor answer often remains high even after the change point, while the probability of predicting the post-change answer increases only weakly. This provides additional evidence of outdated persistence and weak boundary-sensitive adaptation.

D Experimental Details

Model checkpoints. We use LLaMA-3.1-8B-Instruct [3], Qwen3-4B-Instruct [4], and Mistral-7B-Instruct-v0.3 from Hugging Face, and GPT-4o through the official API. All models are evaluated

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

[4] https://huggingface.co/Qwen/Qwen3-4B-Instruct-2507

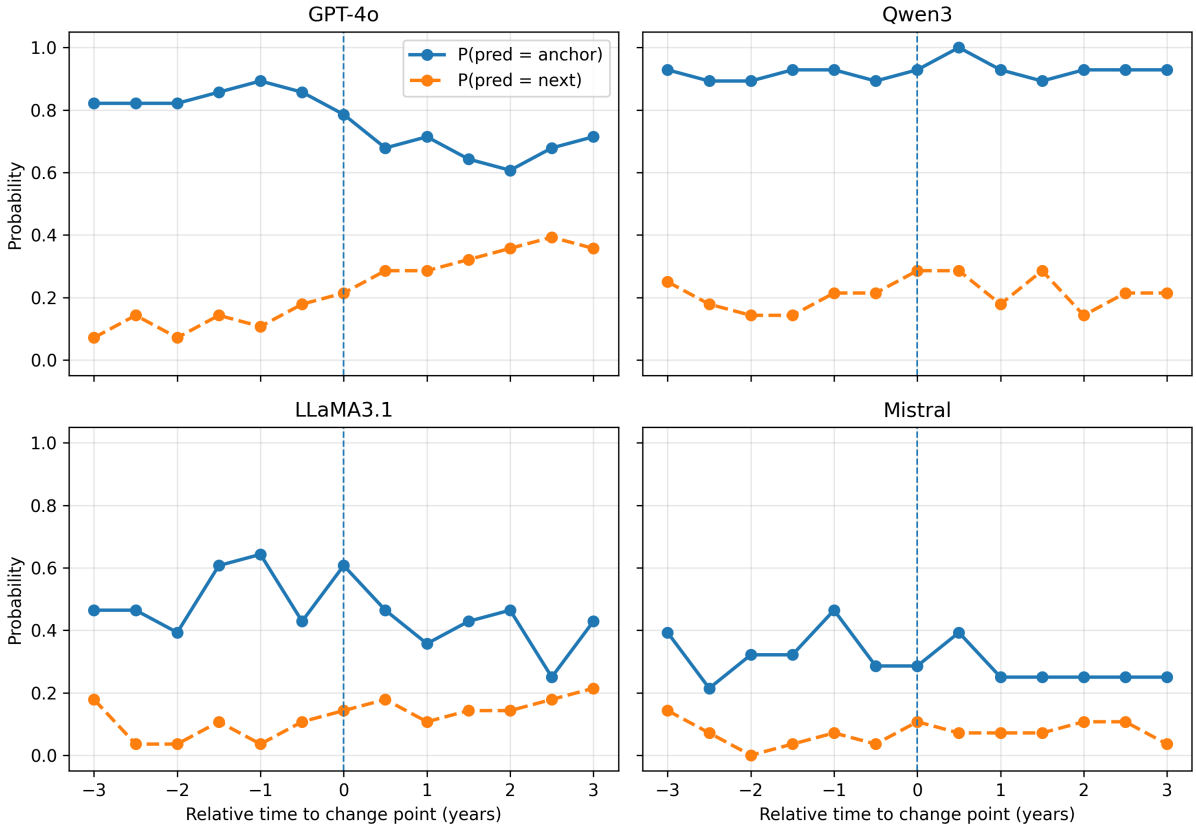

Figure 7: Prediction flip curves around factual change points. The blue curve shows the probability of predicting the pre-change anchor answer, while the orange curve shows the probability of predicting the post-change answer. A boundary-aware model should reduce anchor-answer predictions and increase post-change predictions after the change point.

without browsing, tools, open-corpus retrieval, or access to refreshed gold annotations.

Controlled retrieval baseline. We do not use open-corpus RAG as the baseline, since it would introduce retrieval-specific confounds such as corpus construction, chunking, retriever quality, and temporal leakage from post-hoc updated documents. Both CTR and the baseline operate on the same bounded historical evidence, allowing us to focus on reasoning and commitment behavior rather than retrieval pipeline design.

Implementation. Experiments are run on a Linux server equipped with 4 NVIDIA A100 GPUs with 80GB memory each.

E Case Studies and Confidence-Aware Repair Analysis

We provide additional qualitative analyses to illustrate how CTR improves reliability in future-oriented temporal reasoning. The cases cover three representative failure modes of controlled retrieval reasoning: unsupported extrapolation, factual hallucination, and spurious temporal change detection.

E.1 Unsupported Extrapolation and Factual Hallucination

Table 3 compares the controlled retrieval baseline with CTR on two representative cases. These cases highlight how controlled retrieval reasoning can produce plausible but weakly supported future answers, while CTR regulates commitment based on temporal evolution evidence.

Unsupported extrapolation. In the case concerning Shoshana Zuboff's publications, the retrieval-guided baseline predicts an increase in the number of books without citing concrete historical evidence, reflecting a tendency to assume change by default. CTR, in contrast, evaluates the temporal reliability of historical recall and identifies a long stable period with consistently high confidence. Based on this inferred slow-changing evolution pattern, CTR conservatively maintains the historical value for 2025.

Case	Model	Gold	Controlled Retrieval	Confidence-Aware TR (CTR)
The number of books published by Shoshana Zuboff	GPT-4o	3	**Answer:** 4 **Reasoning:** Assumes a slow publication rate without citing concrete historical evidence.	**Answer:** 3 **Reasoning:** "The number of books remained stable at 3 from 2018 to 2024. Given the *Slow-changing* pattern, it is likely unchanged in 2025."
Children of Leonardo DiCaprio	Qwen3	0	**Answer:** 1 **Reasoning:** "Leonardo DiCaprio has been publicly known to have one child, a daughter named Matilda, born in 2004."(*Hallucination*)	**Answer:** 0 **Reasoning:** "The evidence indicates a stable period from 1978 to 2024 with no children."

Table 3: Two representative case studies comparing the controlled retrieve-then-reason baseline with our CTR framework. The baseline exhibits speculative extrapolation (Case 1) and factual hallucination (Case 2), whereas CTR produces more reliable predictions through structured temporal grounding.

Factual hallucination. In the Leonardo DiCaprio case, the retrieval-guided baseline fabricates a specific but non-existent detail, namely "a daughter named Matilda", showing that LLMs can generate coherent yet unsupported narratives without explicit grounding. CTR avoids this error by aggregating decades of consistent historical evidence. The confidence-aware mechanism filters out low-probability generative noise, allowing the model to conclude that the factual state remains "Zero".

E.2 Confidence-Aware Repair under Noisy Historical Recall

Figure 8 illustrates a representative case for the query "What was the shortest war in history?", highlighting how CTR preserves prediction reliability under noisy historical recall.

In early years, historical recall exhibits substantial fragmentation, alternating between multiple candidates such as the Anglo-Zanzibar War and the Anglo-Portuguese War. Although some recalled answers appear with high instantaneous confidence, the overall confidence signal fluctuates sharply, making these early predictions unreliable.

Around 1991, a deviation to the Anglo-French War is detected as a potential change point. However, this change point receives low weight because it is isolated and lacks sustained support in subsequent years. CTR therefore treats it as a spurious fluctuation rather than genuine factual evolution.

From 1992 onward, the Anglo-Zanzibar War forms a long stable period with consistently high confidence and low variance, establishing a reliable temporal anchor. Based on this stability, CTR commits to the Anglo-Zanzibar War as the future answer.

In contrast, the retrieval-guided baseline produces a fluent but hallucinated reasoning chain by enumerating multiple unrelated conflicts without temporal grounding. This comparison demonstrates that CTR enables conservative future commitment by suppressing unreliable change signals and prioritizing long-term temporal stability.

Overall, these cases show that CTR acts as a reliability layer for future-oriented temporal reasoning: it reduces unsupported extrapolation, avoids factual hallucination, and prevents isolated noisy recalls from being misinterpreted as meaningful temporal changes.

F Reasoning Hallucination Judgement

To ensure the reliability of our hallucination evaluation, we employ a rigorous two-stage verification pipeline, combining broad automated screening with fine-grained human adjudication.

Stage 1: Strict Automated Screening. First, we utilize GPT-5.1 as an independent judge to perform an initial screening of all reasoning traces. The judge is instructed to flag instances based on three strict criteria:

1. **Fabrication:** Citing specific events, names, or data points that do not exist (e.g., inventing a "2025 release date" for a cancelled movie).

2. **Unsupported Extrapolation:** Making definitive claims about volatile future states without citing stable historical patterns or official announcements.

3. **Logical Contradiction:** Deriving a conclusion that contradicts the cited historical evidence within the trace.

This stage acts as a high-recall filter to identify potential hallucinations ($\text{Hallucination}_{i,k} = 1$).

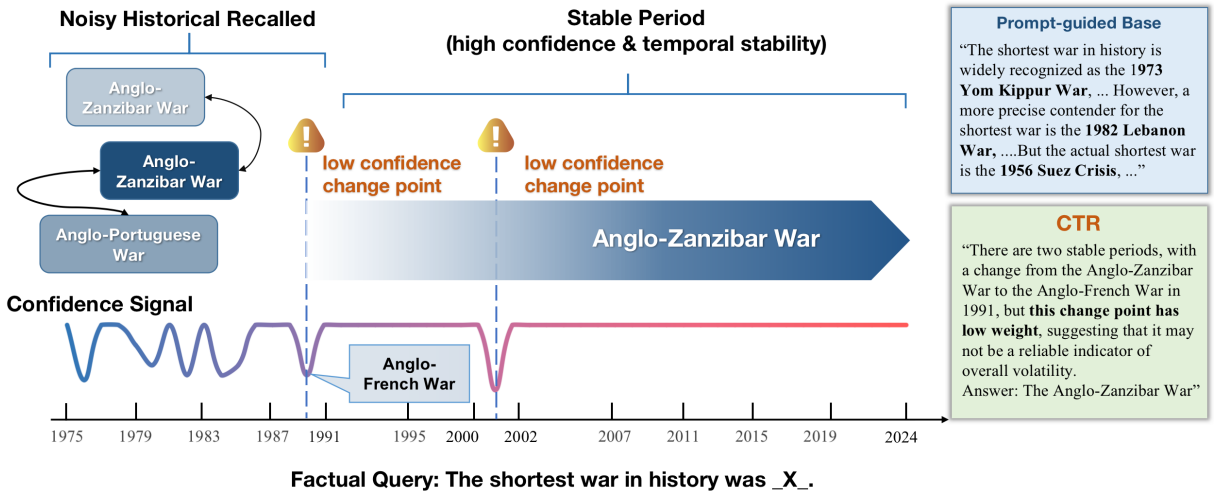

Figure 8: An illustrative case showing how CTR suppresses low-confidence change points and commits to future predictions only when long-term temporal stability is observed.

Stage 2: Expert Human Adjudication. To eliminate false positives from the automated judge and ensure gold-standard quality, we conduct a secondary human review. This process is performed by expert annotators familiar with the temporal reasoning task. The human review protocol involves:

- **Fact Verification via Search:** For every flagged instance, annotators are required to verify the validity of the generated claims against external search engines (e.g., Google Search) to confirm whether the "hallucinated" detail is indeed false or merely obscure.
- **Logic and Grounding Check:** Annotators assess whether the reasoning logically follows from the provided context. A trace is marked as valid only if the prediction is grounded in verifiable patterns (e.g., "consistent 4-year cycle") rather than spurious correlations.
- **Final Adjudication:** In cases of disagreement between the automated judge and human assessment (e.g., the model correctly inferred a complex fact that GPT-5.1 mistook for hallucination), the human judgment prevails.

This human-in-the-loop refinement ensures that our reported hallucination rates reflect genuine generative failures rather than evaluation artifacts.

We emphasize that hallucination rates are used for controlled relative comparison across methods under identical judging conditions, rather than as absolute measures of factuality.

G Dataset Selection and Annotation Details

G.1 Predictability Labeling Protocol

Predictability labels are assigned at the instance level under the bounded-evidence setting. Let E_i denote the historical temporal evidence available for instance i, and t_i^* the queried future time. We label i as PREDICTABLE if E_i supports a unique answer at t_i^* through one of the following patterns:

(i) stable persistence, where the factual state remains unchanged across the relevant horizon;

(ii) a known temporal boundary, where the validity interval of the current or next state is specified;

(iii) a deterministic rule or cycle, such as calendar-based recurrence. If the target value depends on external announcements, future events, or updates not available in E_i, the instance is labeled UNPREDICTABLE.

This criterion is independent of the evolution regime annotation. In the final diagnostic set, FAST instances are selected when they meet the UNPREDICTABLE criterion, whereas SLOW and NEVER instances are selected when they meet the PREDICTABLE criterion. Thus, the observed type-label mapping is induced by our selection protocol and should not be interpreted as a universal claim that all FAST facts are unpredictable or all SLOW/NEVER facts are predictable.

G.2 Boundary Cases

A FAST-changing fact can be labeled PREDICTABLE if the future value is already fixed by bounded evidence, such as an officially

Fact Type	Query (q_i)	Ref. Time (t_0^i)	Current Answer (a_0^i)	Next Answer (a_1^i)
Never	_X_ won the 2020 Formula 1 world driver's championship.	2022	Lewis Hamilton	N/A
Never	The real name of the Unabomber was _X_.	2022	Theodore John Kaczynski	N/A
Slow	The Winter Olympics were held most recently in _X_.	2022	Beijing, China	Italy
Slow	The next leap year is _X_.	2024	2028	2032
Fast	The most recent movie in the Marvel Cinematic Universe is _X_.	2025	The Fantastic Four: First Steps	Spider-Man: Brand New Day
Fast	King Gizzard's most recent studio album is _X_.	2025	Phantom Island	N/A

Table 4: Representative dataset examples. The Next Answer field is used only for evaluation and is not provided to the model. N/A indicates that no annotated future transition answer is provided; its interpretation depends on the evolution type.

announced schedule. Such cases are not included in the FAST subset of our diagnostic benchmark. Conversely, a SLOW-changing fact can be labeled UNPREDICTABLE if the queried time lies beyond a known validity boundary and no successor state is supported by the evidence. These cases are excluded or relabeled during annotation. This filtering ensures that the benchmark evaluates evidence-bounded predictability rather than raw volatility alone.

G.3 Gold Refresh

The gold refresh date is *September 2025*. This means that stale answers inherited from FRESHQA or TEMPLAMA are re-verified and corrected against the world state at that date. The refreshed gold answer is used only as an evaluation label. It is not part of the bounded historical evidence and is never shown to the model. For rule-governed instances, such as calendar questions, the label is computed from the target time t^*.

G.4 Next Answer

The *next_answer* field is not the predictability label. It is recorded only when a future state is externally verifiable and useful for evaluation. For NEVER facts, *N/A* means that no future transition is annotated because the current answer is expected to remain stable. For FAST facts, *N/A* means that no stable or justified future target is assumed under bounded evidence.

H Dataset Examples

Table 4 presents representative examples from our evaluation dataset. Each instance is annotated with its evolution type, cloze-style query, reference time, current answer, and an optional next-answer field. The structure strictly follows the formal definition provided in Section 3, where each instance is annotated as a tuple $e_i = \langle q_i, t_0^i, a_0^i, t_c^i, a_1^i \rangle$. The transition time t_c^i and next answer $a_1^i$are used for evaluation and analysis only, and are not provided to the model during probing or inference.

Specifically, the table columns correspond to:

- **Fact Type**: The evolution category.
- **Query** (q_i): The cloze-style factual query.
- **Ref. Time** (t_0^i): The latest reference time at which the fact is known to be valid (corresponding to the `effective_year` in the raw data).
- **Current Answer** (a_0^i): The correct answer valid at time t_0^i.
- **Next Answer** (a_1^i): The verifiable future answer after the change point. This is set to N/A for *Never-changing* facts or *Fast-changing* facts where the future state is highly volatile or yet to be determined.

The *Current Answer* field denotes the factual answer verified at the reference time. The *Next Answer* field is used only for evaluation and analysis when a future answer is well-defined and verifiable. It is not provided to the model during inference.

Importantly, N/A in the Next Answer field should not be interpreted uniformly as an unpredictable future state. For NEVER-changing facts, N/A means that no future transition is annotated because the current answer is expected to remain stable. For FAST-changing facts, N/A indicates that no stable or justified future answer is assumed under the available evidence. Thus, evaluation decisions depend on both the evolution type and the availability of a verifiable future answer.