AnRe: Analogical Replay for Temporal Knowledge Graph Forecasting

Guo Tang[1], **Zheng Chu**[1], **Wenxiang Zheng**[1], **Junjia Xiang**[1], **Yizhuo Li**[1], **Weihao Zhang**[1], **Ming Liu**[1,2*], **Bing Qin**[1,2]
[1]Harbin Institute of Technology, Harbin, China
[2]Peng Cheng Laboratory, Shenzhen, China
{gtang,zchu,wxzheng,mliu,qinb}@ir.hit.edu.cn

Abstract

Temporal Knowledge Graphs (TKGs) are vital for event prediction, yet current methods face limitations. Graph neural networks mainly depend on structural information, often overlooking semantic understanding and requiring high computational costs. Meanwhile, Large Language Models (LLMs) support zero-shot reasoning but lack sufficient capabilities to grasp the laws of historical event development. To tackle these challenges, we introduce a training-free **An**alogical **Re**play (**AnRe**) reasoning framework. Our approach retrieves similar events for queries through semantic-driven clustering and builds comprehensive historical contexts using a dual history extraction module that integrates long-term and short-term history. It then uses LLMs to generate analogical reasoning examples as contextual inputs, enabling the model to deeply understand historical patterns of similar events and improve its ability to predict unknown ones. Our experiments on four benchmarks show that AnRe significantly exceeds traditional training and existing LLM-based methods. Further ablation studies also confirm the effectiveness of the dual history extraction and analogical replay mechanisms.

1 Introduction

As a structured tool for representing real-world facts, Temporal Knowledge Graphs (TKGs) extend traditional knowledge graphs by incorporating temporal information, thereby capturing the dynamic evolution of knowledge (Leblay and Chekol, 2018; Garcia-Duran et al., 2018; Xiang et al., 2022; Chen et al., 2023). They play a pivotal theoretical role in various research domains such as recommendation systems (Wang et al., 2019), information retrieval (Liu et al., 2018) and social crisis early warning systems (Gastinger et al., 2023).

Early research primarily focuses on training graph neural networks (GNNs) and recurring neural networks (RNNs) to model the relationships between entities and relations in TKGs (García-Durán et al., 2018; Zhu et al., 2021; Han et al., 2021; Li et al., 2022; Xu et al., 2023b). Although some methods (Jin et al., 2019; Li et al., 2021) attempts to predict recurring or periodic events by referencing known events, they often fall short in effectively modeling semantic information and require substantial computational resources for training on specific datasets.

Recent studies explore LLMs for reasoning on TKGs (Yang et al., 2023; Yuan et al., 2024; Lee et al., 2023). For instance, Lee et al. (2023) proposes an In-Context Learning (ICL) approach, providing semantic information of TKGs to LLMs in textual form. Yu et al. (2024) employs dynamic rule libraries to present relevant historical context to LLMs. Xia et al. (2024a) enhances the historical context by constructing higher-order historical

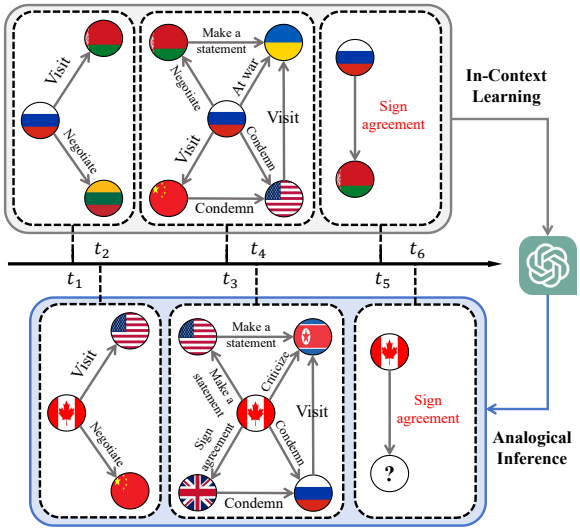

Figure 1: An example of inference on TKG using AnRe. Leveraging the in-context learning capabilities of LLMs, the model is able to effectively predict outcomes for unseen queries after learning similar historical development processes.

* Corresponding Author.

Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 4632–4650

inputs for LLMs. Although these methods eliminate the need for fine-tuning, they still face the following issues: First, they rely solely on zero-shot inference, neglecting the exceptional learning capabilities of LLMs from reasoning examples. Second, most methods fail to verbalize events into natural language sentences, resulting in insufficient semantic coherence in the historical context, which hinders the LLM's ability to comprehend complex historical backgrounds effectively.

To address these, our research focuses on leveraging the semantic understanding and few-shot learning capabilities of LLMs (Zhao et al., 2023; Brown et al., 2020) to enhance predictive performance and reduce computational resource consumption. We propose a training-free **An**alogical **Re**play (**AnRe**) framework for TKGF tasks, as illustrated in Figure 1. Given a query *(Canada, Sign agreement, ?, t_6)*, we identify a similar event *(Russia, Sign agreement, Belarus, t_5)*. By learning the historical development of this similar event, the model can perform analogical reasoning on the unknown query, thereby significantly improving prediction accuracy.

Our framework consists of three core modules:(i) *Semantic-driven Historical Clustering*: We transform entities into semantic vectors and cluster them to identify entities semantically similar to the target entity and events analogous to the given query. (ii) *Dual History Extraction*: We verbalize the historical events and queries, originally in tuple form, into sentences of natural language incorporating appropriate connecting words. Combining an improved long-term and short-term history extraction approach, we construct similar queries by masking the object in similar events and extract short-term history based on temporal proximity. Simultaneously, we compute the probability distribution of the validity of historical events using the semantic understanding capabilities of LLMs, dynamically filtering out effective long-term history. (iii) *Analogical Replay*: We utilize LLMs to construct the reasoning process for similar queries and their historical contexts, generating analogical reasoning examples. The model learns the historical development process through in-context learning and ultimately computes the probability distribution of candidate objects for the target query, selecting the most likely prediction result.

We conduct extensive experiments on widely used TKG benchmarks, including ICEWS14 (Garcia-Duran et al., 2018), ICEWS05-15 (Garcia-Duran et al., 2018), ICEWS18 (Jin et al., 2019) and GDELT (Leetaru and Schrodt, 2013). We use InternLM2 (Cai et al., 2024b), Qwen2.5 (Yang et al., 2024; Team, 2024), Yi (AI et al., 2024), Mistral (Jiang et al., 2023a), and Llama-3 (AI@Meta, 2024) in our experiments. The results demonstrate that our method significantly outperforms trained baseline methods on most metrics across four datasets. Among methods employing LLMs, AnRe achieves the best performance across various LLMs. Compared to untrained methods, our method shows average improvements of 17.8% and 9.4%, respectively, proving the effectiveness of our approach. Furthermore, ablation studies and hyperparameter analysis highlight the advantages of our dual history extraction and analogical replay mechanisms.

Our contributions are summarized as follows:

(1) We propose an Analogical Replay (AnRe) reasoning framework for TKGF, which leverages the few-shot learning capabilities of LLMs to enhance predictive performance.

(2) We innovatively introduce a process for constructing analogical examples and a dual history extraction method, providing LLMs with efficient analogical replay contexts.

(3) Our experiments demonstrate the effectiveness of AnRe in various datasets and metrics, showcasing significant advantages compared to the baseline methods.

2 Task Formulation

2.1 Temporal Knowledge Graph Forecasting

Temporal Knowledge Graph Forecasting (TKGF) involves predicting future states of a TKG, which is a series of time-ordered multi-relational directed graphs. Formally, a TKG up to time t is denoted as $\mathcal{TKG}_t = \{G_1, G_2, \ldots, G_t\}$, where each snapshot $G_t = (\mathcal{V}, \mathcal{R}, \mathcal{E}_t)$ represents the graph at time t. Here, $\mathcal{V}$ is the set of entities, $\mathcal{R}$ is the set of relations, and $\mathcal{E}_t$ consists of timestamped events e as quadruples (s, r, o, t) with $s, o \in \mathcal{V}$ and $r \in \mathcal{R}$.

The goal of TKGF is to predict missing entities in a quadruple for a future time $t + k$, either as an object $(s, r, ?, t + k)$ or a subject $(?, r, o, t + k)$, leveraging historical data from previous snapshots $\mathcal{TKG}_{<t} = \{G_1, G_2, \ldots, G_{t-1}\}$. Candidate entities $e_i \in \mathcal{V}$ are assigned scores by TKG prediction models to determine the unknown entities.

2.2 In-Context Learning for TKGF

In-Context Learning (ICL) for TKGF leverages large language models (LLMs) to adapt to forecasting tasks using contextual examples, eliminating the need for model fine-tuning (Lee et al., 2023). In this approach, ICL uses historical data from a temporal knowledge graph to predict future events. For a future query $q = (s_q, r_q, ?, t_n)$, where s_q is an entity and r_q is a relation at time t_n, the method retrieves the historical context $H_n(q)$ from previous graph snapshots $\mathcal{TKG}_{1:n-1} = \{G_1, \ldots, G_{n-1}\}$.

A prompt θ_q is constructed based on this historical context. The prediction y_q is then generated from the LLM's probability distribution $y_q \sim P_{\text{LLM}}(y_q \mid \theta_q)$ using ICL to make zero-shot predictions. Entities and relations are numerically mapped to handle multi-word names, ensuring consistency in the LLM's outputs. This process involves ranking candidate entities using token probabilities without additional training, allowing effective zero-shot forecasting.

3 Analogical Replay Reasoning

The AnRe framework relies primarily on the semantic comprehension capabilities of LLMs (Zhao et al., 2023) and their ability to learn from few-shot examples for inference (Yu et al., 2020; Cobbe et al., 2021; Wei et al., 2022; Kojima et al., 2022). We verbalize the quadruples in the TKG into natural language form, employ semantic clustering to identify similar events, and construct analogous historical development processes. This facilitates the model's ability to infer unknown queries by revisiting past events in unfamiliar ICL tasks. We implement a dual retrieval strategy that integrates both short-term and long-term histories to effectively incorporate causal information.

As illustrated in Figure 2, AnRe operates as follows: Initially, we employ a semantic-driven approach to cluster entities and histories based on the given query. Subsequently, using the LLM, we retrieve a combined long- and short-term history in reverse chronological order. Finally, we utilize the same LLM to perform analogical replay for inference, thus obtaining the predicted entity results.The Algorithm of the whole procedure can be referred in Appendix A. We will introduce semantic-driven historical clustering in § 3.1, dual history extraction in § 3.2 and analogical replay in § 3.3.

3.1 Semantic-driven Historical Clustering

The progression of history often exhibits similarities. For event prediction tasks concerning a specific entity, we frequently infer the development of unknown events by observing how similar events of analogous entities have unfolded historically. The purpose of this module is to employ a semantic-driven clustering approach to identify historically analogous known events for the masked query. Additionally, it aims to retrieve relevant histories and candidate prediction answers for the query, and relevant histories corresponding to each similar event.

Entity Semantic Clustering For a given query $q = (s_q, r_q, ?, t_{n+1})$, we need to identify the set of similar entities for the target entity s_q. For the entity set in the TKG (represented as $\mathcal{V}$) , we first convert them into vector representations using BERT (Devlin, 2018). We then determine the optimal number of clusters by employing the elbow method and silhouette coefficient (Rousseeuw, 1987). The cluster number k with the highest silhouette coefficient is selected for the final k-means clustering (Lloyd, 1982) of the vector representations. Subsequently, we retrieve the cluster $\mathcal{X}$ to which s_q belongs and consider the set of entities within $\mathcal{X}$ as the semantically similar entities to s_q.

Candidate History Filter In the candidate history filter phase, we first transform all quadruples in $\mathcal{TKG}_{t<n+1}$ by adding prepositions to construct sentences comprehensible to the LLM. Next, we identify similar events and the relevant corresponding histories for the semantically similar entities in $\mathcal{X}$. For an entity s_i in $\mathcal{X}$ and the given timestamp t_{n+1}, we define the relevant historical context as: $H_i = \{(s, r, o, t) \in \mathcal{TKG}_{t<n+1} \mid s = s_i \text{ or } o = s_i\}$. For entities s_i in $\mathcal{X}$ other than s_q, we define the set of similar events as: $E_i = \{(s_i, r, o, t) \in \mathcal{TKG}_{t<n+1} \mid r = r_q\}$.

To identify similar events with sufficient inference information, we filter E_i by requiring at least 300 relevant historical contexts before the event timestamp. These events are then ranked according to the semantic similarity to q, and the query with the highest similarity is selected as the similar event e_i. Since these similar events are retrieved before t_{n+1}, their corresponding values o are known.

For the target entity s_q, we first identify the relevant history H_q using the same method, and then determine the candidate answer entity set O_q for query q. We consider entities that have interacted

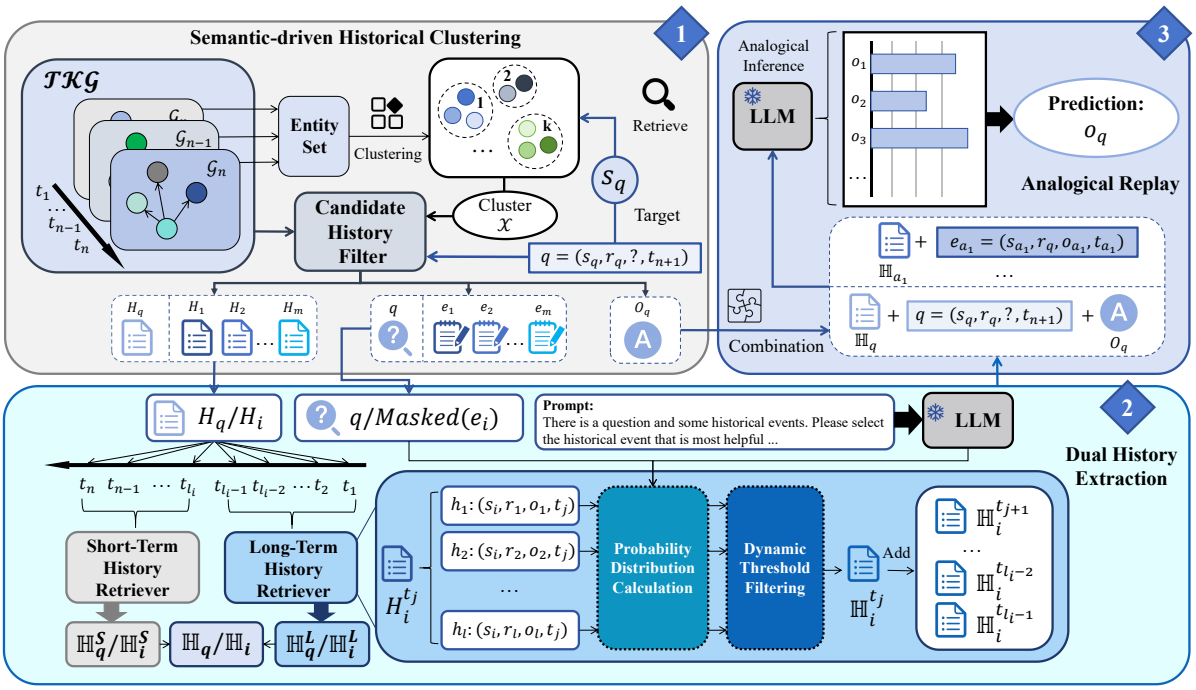

Figure 2: Overview of AnRe. (1) Semantic-driven Historical Clustering: Clustering TKG entities and filtering event sets by query. (2) Dual History Extraction: Retrieving long-term and short-term histories with dynamic filtering. (3) Analogical Replay: Constructing ICL prompts for analogical reasoning and entity prediction. It is noteworthy that the quadruples in the TKG are verbalized into natural language sentences.

with s_q in a large number of historical events as potential candidates, thus defining the candidate answer entity set as: $O_q = \{o \mid (s, r, o, t) \in H_q, s = s_q\} \cup \{s \mid (s, r, o, t) \in H_q, o = s_q\}$.

3.2 Dual History Extraction

Utilizing a combined input of long- and short-term histories allows for the capture of historical context information across different time scales. The short-term history provides the immediate context of the query event, which empirically may have a direct impact on the query and can capture immediate variability and abrupt changes. The long-term history aids in identifying long-term trends and patterns of event occurrences, offering a broader temporal macro-context. Unlike ONSEP (Yu et al., 2024), which predicts separately for long-term and short-term histories and then weights the results, we integrate the long-term and short-term histories into a single historical context, providing the model with a more coherent and semantically rich background.

3.2.1 Short-Term History Retriever

The short-term history retriever focuses on acquiring the most recent event chains that are close in time to the query or similar events. For the relevant history H_i corresponding to a similar event e_i, we sort the events by their timestamps and truncate the most recent l events to form the relevant short-term history chain $\mathbb{H}_i^S$. Similarly, we truncate the relevant history H_q for the query q to form the query's short-term history chain $\mathbb{H}_q^S$. The short-term history chain is directly related to the entities in the query or similar events and may have direct causal relationships, making it a crucial reference for prediction.

3.2.2 Long-Term History Retriever

The long-term history retriever focuses on retrieving event chains across extended time spans. This stage primarily consists of two modules: Probability Distribution Calculation (PDC) and Dynamic Threshold Filtering (DTF). For a similar event e_i, the long-term history chain is obtained from the set $H_i^L = H_i - \mathbb{H}_i^S$. We partition the set H_i^L into historical sets for each time step, denoted as: $H_i^{t_j} = \{(s_i, r, o, t_j) \mid (s_i, r, o, t_j) \in H_i^L, t_{l_i} > t_j \geq t_1\}$, where t_{l_i} is the timestamp of the last event in $\mathbb{H}_i^S$. For the query q, we employ the same method to obtain the historical sets $H_q^{t_j}$. At each time step, we employ PDC and DTF to complete the long-term history retrieval.

Probability Distribution Calculation leverages the powerful semantic comprehension capabilities of the LLM to assess the effectiveness of history for query inference. We first mask the entity o in the similar event e_i to construct a query q_i with the same structure as q. Next, we specify a structured prompt θ_1 (Table 7) and, at each time step t_j, instruct the model to identify the most helpful historical events from $H_i^{t_j}$ for inferring q_i. We numerically map each historical event in $H_i^{t_j}$, utilize the LLM to obtain the corresponding logarithmic output L_p, and then convert it into a normalized probability using the softmax function. For $H_q^{t_j}$, we calculate the probability distribution using the same method. The result represents the LLM's judgment on the effectiveness of each historical event as an inference context.

Specifically, for any historical event h_l in $H_i^{t_j}$, we first map it to a label id_l. We then obtain the logarithmic scores $s = \text{LLM}(\theta_1(H_i^{t_j}, q_i))$ for each historical event. The effectiveness probability distribution for each historical event is calculated as follows:

$$p(h_l) = \frac{e^{s_{id_l}}}{\sum_{h_k \in H_i^{t_j}} e^{s_{id_k}}} \tag{1}$$

Dynamic Threshold Filtering primarily filters out effective histories for each time step by setting dynamic confidence thresholds. Empirically, the correlation between historical events and the query diminishes with increasing temporal distance from the query. Therefore, we require higher probability confidence to ensure the model's grasp of event effectiveness. We construct a dynamic threshold calculation method using the time difference between the history and the query. Specifically, for the query q_i, we define the maximum time difference $T = t_{q_i} - t_1$, and the time difference between the history at time step t_j and the query as $\Delta t = t_{q_i} - t_j$. Given the set size F of $H_i^{t_j}$, the confidence threshold should be no lower than the average probability $1/F$ and no higher than 1. Thus, the dynamic threshold c for time step t_j is:

$$c_j = \frac{1}{F} + \left(1 - \frac{1}{F}\right)\left(\frac{\Delta t}{T}\right)^{\alpha} \tag{2}$$

where the variation factor α controls the growth rate of the confidence threshold. When the effectiveness probability $p(h_l) \geq c_j$, h_l is considered a strongly correlated event with q_i and is included in the long-term history set $\mathbb{H}_i^{t_j}$ for time step t_j.

We start retrieving the long-term history sets from time step t_{l_i-1} in reverse chronological order until the long-term history length is sufficient, and then sort them in chronological order to obtain the long-term history chain $\mathbb{H}_i^L$ for q_i. Finally, we concatenate $\mathbb{H}_i^S$ and $\mathbb{H}_i^L$ to obtain the combined long-term and short-term history chain $\mathbb{H}_i$. For q, we use the same method to obtain $\mathbb{H}_q$.

Dataset	# Entity	# Relation	Train	Valid	Test
ICEWS14	12,498	260	323,895	-	341,409
ICEWS05-15	10,094	251	368,868	46,302	46,159
ICEWS18	23,033	256	373,018	45,995	49,545
GDELT	7,691	240	1,734,399	238,765	305,241

Table 1: Statistics of the datasets.

3.3 Analogical Replay

The analogical replay phase aims to leverage the LLM's few-shot learning capabilities to learn from the development processes of similar events, providing insights for inferring unknown queries. We propose a novel analogical replay method for TKGF. Specifically, we filter out e_i from $\mathbb{H}_i$ if its length is less than the total length L, and select the a most semantically similar events to q from the remaining similar events to construct analogical examples. We formulate a structured prompt θ_2 (Table 8) to obtain the LLM's analysis process p for each similar event e_{a_i}, denoted as $p_{a_i} = \text{LLM}(\theta_2(\mathbb{H}_{a_i}, e_{a_i}))$. The analogical example ex_{a_i} is constructed as $(\mathbb{H}_{a_i}, e_{a_i}, p_{a_i})$.

The a analogical examples form the set $\mathcal{P}$, which, combined with $\mathbb{H}_q$, q, and O_q, is used to construct a structured prompt θ_3 (Table 9). The same LLM is utilized to select the tail entity o from the candidate entity set O_q for prediction. Similar to the PDC phase, we map each candidate entity to a numerical token, obtain the corresponding logarithmic output L_a from the LLM, and convert it into a normalized probability using the softmax function, resulting in the probability distribution of each candidate answer. We sort the probability results and select the highest probability result as the final prediction.

4 Experimental Setup

4.1 Datasets and Evaluation Metrics

We conduct experiments on two widely-used event datasets in the TKGF domain: the Integrated

Crisis Early Warning System (ICEWS) dataset, specifically using ICEWS14 (García-Durán et al., 2018), ICEWS05-15 (García-Durán et al., 2018), and ICEWS18 (Jin et al., 2019) versions, and the Global Database of Events, Language, and Tone (GDELT) dataset (Leetaru and Schrodt, 2013). Table 1 presents the statistics of the datasets. These datasets represent political events in the form of quadruples, such as *(Barack Obama, visit, Malay, 2014/02/19)*, where each event consists of a subject, relation, object, and timestamp. We construct historical event chains using the training sets of these datasets and randomly select 200 events from the test sets for experimental evaluation, averaging the results of three experiments as the final outcome. To assess the effectiveness of our method in predicting rankings, we employ the Hit@k (where k = 1, 3, 10) evaluation metrics, which measure the accuracy of our method in the top k predictions.

4.2 Baselines

For traditional supervised models, we select RE-NET (Jin et al., 2019), xERTE (Han et al., 2020b), CyGNet (Zhu et al., 2021), RE-GCN (Li et al., 2021), TITer (Sun et al., 2021), TiRGN (Li et al., 2022) and DiffuTKG (Cai et al., 2024a) for performance comparison. Among methods utilizing LLMs, we primarily compare our approach with ICL (Lee et al., 2023), ONSEP (Yu et al., 2024), CoH (Xia et al., 2024a) and LLM-DA (Wang et al., 2024). Additionally, we evaluate the performance differences using different LLMs within our framework. The details of LLMs used in experiments can be found in Appendix B.

4.3 Implementation Details

We convert the structured quadruples within the TKG into textual sentences by incorporating prepositions for our experimental purposes. Sample textual inference examples are furnished in Appendix F. Recognizing the influence of input length on the textualized data, all experiments are executed on an NVIDIA A100 Tensor Core GPU equipped with 80GB of VRAM.

5 Experimental Results

5.1 Main Results

As shown in Table 2, we selected several traditional embedding-based models, hybrid models combining graph-based approaches with LLMs, and purely LLM-based methods for comparison. Across all four datasets, our method demonstrates strong competitiveness. Among the purely LLM-based approaches (with the history length L set to 100), our framework achieves the best performance. By expanding the candidate entity set to include the second-order historical neighbors of s_q, we observe a slight yet consistent performance improvement. Compared to ICL, our method achieves improvements of 22.59%, 10.76%, and 51.16% in Hit@1, while showing gains of 11.82%, 1.30%, and 30.00% over ONSEP. When compared to graph-based methods, our model outperforms most baseline approaches and remains competitive with SOTA methods in certain metrics. Moreover, our approach provides a better balance between computational efficiency and predictive performance compared to the training overhead required by graph-based models.

5.2 Ablation Study

We conduct three sets of ablation experiments on four benchmarks, with the results summarized in Table 3.

Effect of Analogical Examples To investigate the impact of analogical reasoning examples, we remove these examples when constructing the final inference prompt, requiring the model to directly learn from the historical event chain and predict the outcome. The results show a significant decline in model performance after removing the analogical examples. This indicates that after learning the development process of similar historical events, the model can better understand the occurrence patterns of historical events and make predictions. However, even after removing the analogical examples, our method still outperforms ONSEP on multiple metrics, demonstrating the effectiveness of our dual-history combined reasoning approach.

Effect of Dual History To investigate the impact of short-term and long-term histories on prediction, we separately remove the short-term and long-term histories. The results show that after removing the short-term history, the model's prediction performance significantly declines, with the accuracy of Hit@1 dropping by 9.98% on average compared to the original method. This indicates the critical role of short-term history in event prediction, as it provides direct causes or reasoning evidence for the occurrence of queried events.

After removing the long-term history, the performance also declines noticeably, but the impact is

Type	Model	Train	ICEWS14				ICEWS05-15				ICEWS18				GDELT			
			MRR	Hit@1	Hit@3	Hit@10	MRR	Hit@1	Hit@3	Hit@10	MRR	Hit@1	Hit@3	Hit@10	MRR	Hit@1	Hit@3	Hit@10
♣	RE-NET	√	0.399	0.301	0.440	0.582	0.437	0.336	0.488	0.627	0.298	0.197	0.326	0.485	0.196	0.124	0.221	0.340
	xERTE	√	0.408	0.327	0.457	0.573	0.466	0.378	0.523	0.639	0.293	0.210	0.335	0.465	0.195	0.119	0.220	0.342
	CyGNet	√	0.381	0.274	0.426	0.579	0.413	0.294	0.461	0.616	0.278	0.172	0.310	0.469	0.190	0.117	0.219	0.334
	RE-GCN	√	0.420	0.316	0.472	0.617	0.480	0.373	0.539	0.685	0.326	0.224	0.368	0.527	0.197	0.125	0.223	0.338
	TITer	√	0.418	0.328	0.465	0.584	0.476	0.383	0.528	0.649	0.317	0.221	0.335	0.448	0.195	0.127	0.220	0.331
	TiRGN	√	0.429	0.321	0.485	0.636	0.485	0.369	0.552	0.703	0.320	0.210	0.367	0.537	0.217	0.137	0.241	0.376
	DiffuTKG	√	**0.485**	**0.364**	**0.494**	**0.727**	**0.527**	**0.403**	**0.602**	**0.759**	**0.367**	**0.257**	**0.388**	**0.578**	**0.251**	**0.163**	**0.275**	**0.423**
♢	CoH	×	0.439	0.331	0.496	0.649	0.497	0.380	0.564	0.713	**0.330**	**0.218**	**0.378**	**0.549**	-	-	-	-
	LLM-DA	×	**0.471**	**0.369**	**0.526**	**0.671**	**0.521**	**0.416**	**0.586**	**0.728**	-	-	-	-	-	-	-	-
♠	ICL*	×	0.318	0.301	0.432	0.560	0.353	0.353	0.507	0.647	0.215	0.172	0.289	0.434	-	-	-	-
	ONSEP	×	-	0.330	0.464	0.570	-	0.386	0.546	0.662	-	0.200	0.324	0.443	-	-	-	-
	AnRe (O_q)	×	0.466	0.346	0.470	0.608	0.498	0.389	0.551	0.678	0.321	0.255	0.371	0.554	0.221	0.153	0.244	0.342
	AnRe (O_q^2)	×	**0.474**	**0.369**	**0.511**	**0.657**	**0.509**	**0.391**	**0.580**	**0.696**	**0.355**	**0.260**	**0.392**	**0.567**	**0.243**	**0.166**	**0.266**	**0.375**
	*ΔImprove**		49.1%	22.6%	18.3%	17.3%	44.2%	10.8%	14.4%	7.6%	65.1%	51.2%	35.6%	30.7%	-	-	-	-
	ΔImprove		-	11.8%	10.1%	15.3%	-	1.3%	6.2%	5.1%	-	30.0%	21.0%	28.0%	-	-	-	-

Table 2: The comparative performance of traditional embedding-based models (represented as ♣), hybrid models (represented as ♢) and LLM-based prediction models (represented as ♠), evaluated using MRR and Hit@k metrics across four datasets. All LLM-based methods utilize InternLM2-7B (Cai et al., 2024b) as the foundational model. The notation O_q^2 denotes using the set of historical second-order neighbor entities of s_q as the candidate set. The best performance within each model type is highlighted in **bold**. $\Delta Improve*$ and $\Delta Improve$ denote the improvements of our method over ICL and ONSEP, respectively. Results for embedding-based models and hybrid models are excerpted from (Li et al., 2022; Cai et al., 2024a; Wang et al., 2024), while results for LLM-based models are excerpted from (Yu et al., 2024).

Method	ICEWS14				ICEWS05-15				ICEWS18				GDELT			
	MRR	Hit@1	Hit@3	Hit@10	MRR	Hit@1	Hit@3	Hit@10	MRR	Hit@1	Hit@3	Hit@10	MRR	Hit@1	Hit@3	Hit@10
AnRe	**0.466**	**0.346**	**0.470**	**0.608**	**0.498**	**0.389**	**0.551**	**0.678**	**0.321**	**0.255**	**0.371**	**0.554**	**0.221**	**0.153**	**0.244**	**0.342**
w/o example	0.401	0.300	0.431	0.587	0.422	0.347	0.520	0.661	0.288	0.220	0.364	0.513	0.195	0.117	0.215	0.329
w/o short-term	0.353	0.214	0.378	0.533	0.349	0.298	0.476	0.575	0.200	0.121	0.235	0.481	0.175	0.111	0.208	0.307
w/o long-term	0.388	0.241	0.382	0.543	0.401	0.332	0.497	0.603	0.256	0.146	0.255	0.477	0.190	0.114	0.208	0.310

Table 3: Ablation study of AnRe. W/o example: exclude analogical reasoning examples. W/o short-term: exclude short-term history. W/o long-term: exclude long-term history.

less pronounced compared to the absence of short-term history. Compared to ONSEP, the lack of long-term history leads to performance lags in Hit@1 and Hit@3, but the difference in Hit@10 is negligible, with a lead of 0.034 on ICEWS18. This not only demonstrates the effectiveness of our method in coarse-grained prediction but also highlights the significant influence of long-term history on prediction accuracy. In other words, providing a complete historical context over a long time span is crucial.

5.3 Performance Comparison of Different LLMs

We present the complete results of different LLMs in Table 5. As shown in Figure 7, our method achieves the best performance across various large models, demonstrating a significant improvement over ONSEP. Notably, Qwen2.5 performs the best, achieving a Hit@1 result of 0.261 on the ICEWS18 dataset. Compared to InternLM2, Yi and Llama3 shows slight improvements across all three methods. Mistral demonstrates progress in the ICL method but exhibits some shortcomings in the other methods. On the other two datasets, Qwen2.5 and Yi particularly stand out when using our method, with Qwen2.5 achieving the best performance across multiple metrics. These results indicate that AnResignificantly enhances the prediction accuracy of models and exhibits positive effects across multiple LLMs.

6 Analysis and Discussion

6.1 Hyperparameter Analysis

To identify the optimal hyperparameter settings and investigate whether the impact of hyperparameters is consistent across different datasets, we conducted comparative experiments on all hyperparameters used in AnRe.

Historical Length We fix other hyperparameters and conduct experiments under different historical lengths, with the evaluation results shown in

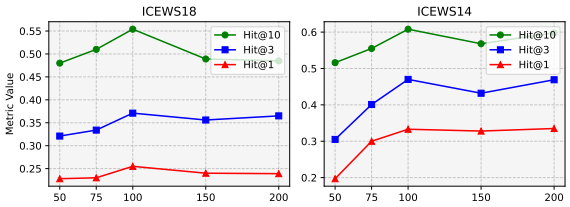

Figure 3: Performance of different historical lengths L on ICEWS14 and ICEWS18.

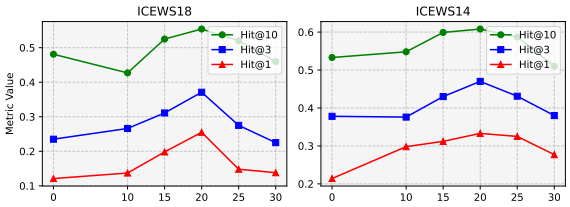

Figure 4: Performance of different short-term history lengths l on ICEWS14 and ICEWS18.

Figure 3. The experimental results indicate that the impact of historical length L on performance follows a similar trend across different datasets. Specifically, when the historical length increases from 50 to 100, the performance of our method continues to improve, peaking at $L = 100$. However, as L grows from 100 to 150, performance declines, with a slight recovery at 200. This suggests that our method still requires a sufficient amount of historical information for reasoning, but excessively long contexts can lead to a decline in the model's prediction capability. Given that our structured prompts require both analogical reasoning history and formal reasoning context, the input length provided to the LLM is longer compared to the ONSEP framework. Consequently, as the historical length increases, the model becomes more susceptible to noise in the historical data, leading to increased volatility in the prediction results.

Short-Term History Length We investigate the impact of different short-term historical lengths l on prediction performance, with the results shown in Figure 4. Specifically, as l increases from 0 to 20, performance generally improves, reaching its peak at $l = 20$ on both datasets. When l increases from 20 to 30, performance gradually declines. This indicates that the impact of short-term historical changes is similar across different datasets, and maintaining an appropriate short-term historical length is crucial for prediction. When l is too short, the model cannot fully utilize key information from recent data, leading to insufficient description of the current state, and the model may struggle to capture short-term fluctuations and trend changes in the data. Conversely, when l is too long, although the model can leverage more historical data, it also introduces excessive redundant information and noise. Therefore, the optimal selection of short-term historical length requires finding a balance between the amount of information and noise.

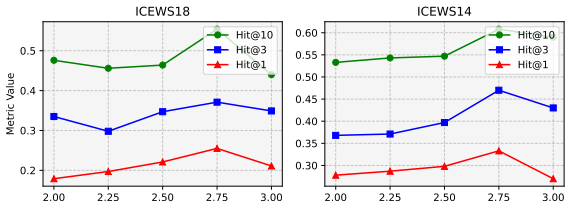

Figure 5: Performance of different threshold variation factors α on ICEWS14 and ICEWS18.

Threshold Variation Factor We investigate the impact of the threshold variation factor α on performance, as illustrated in Figure 5. It is observed that as α increases from 2 to 2.75, the predictive performance of the model across various datasets generally exhibits an upward trend, with the optimal performance achieved at $\alpha = 2.75$. Beyond this value, performance declines. This demonstrates that the influence of α is consistent across different datasets, and selecting an appropriate value allows for the effective retrieval of historical events that are beneficial for prediction. Specifically, when α is too small, the confidence threshold increases too rapidly with time steps, leading to overly stringent filtering of historical events and resulting in insufficient information. Conversely, when α is too large, the confidence threshold remains low, introducing excessive redundant information that hinders the model's capacity to accurately capture key events.

6.2 Analysis of Candidate Set

Our analysis of candidate selection reveals a critical trade-off between recall and efficiency in constructing the candidate set O_q. While the 1-hop neighbor restriction in O_q ensures token efficiency, it also introduces a recall limitation, as evidenced by the lower probability (65–72% across datasets) of the correct entity appearing in O_q. Expanding to 2-hop neighbors (O_q^2) significantly improves recall to 87–91%, as shown in Figure 6, but at the cost of increased computational overhead due to pairwise time checks and context window constraints. This trade-off underscores the practical challenges of

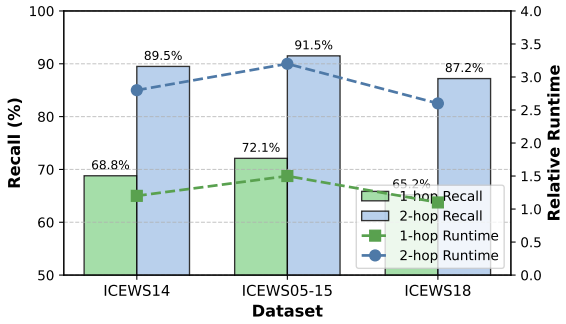

Figure 6: Recall rates and relative runtime for different neighborhood depths.

balancing recall with the operational demands of LLM-based inference. In light of these findings, we adopt O_q^2 in our implementation to mitigate recall limitations while acknowledging the inherent efficiency compromises.

6.3 Analysis of Model Scales

To investigate the performance of models with different scales on our method, we conduct comparative experiments using InternLM2 with 7B and 20B parameters. A detailed analysis can be found in Appendix C.

7 Related Work

7.1 TKGF with Traditional Supervised Models

Traditional TKGF methods typically combine graph-based and sequential models to predict future events. Know-Evolve (Trivedi et al., 2017) and GHNN (Han et al., 2020b) model event evolution through temporal point processes, while CyGNet (Zhu et al., 2021) uses a replication mechanism based on historical event structures. Other models such as RE-Net (Jin et al., 2019) and RE-GCN (Li et al., 2021) integrate GNN to capture both temporal and structural dependencies. TANGO (Han et al., 2021) models continuous-time information using neural ODEs. These methods excel in periodic or repetitive patterns but face limitations in interpretability and zero-shot scenarios. Some recent approaches, such as xERTE (Han et al., 2020a) and MetaTKG++ (Xia et al., 2024b), improve interpretability, but are application-specific. Reinforcement learning and contrastive learning methods (Sun et al., 2021; Xu et al., 2023b) also improve performance, though they struggle with large search spaces and rely on static rules.

7.2 TKGF with LLMs

In contrast, LLMs, with their robust semantic understanding and reasoning capabilities, present a promising avenue for TKG prediction. Initial research (Peters et al., 2019; Han et al., 2023; Yang et al., 2023; Xu et al., 2023a) utilize pre-trained language models (PLMs) to process temporal knowledge by translating historical events into textual contexts for embedding extraction. Subsequent studies (Jiang et al., 2023b; Yuan et al., 2024; Tan et al., 2023) have further explored the integration of temporal and structural information within LLMs. Notably, zrLLM (Ding et al., 2024) and LLM-DA (Ding et al., 2024) have advanced zero-shot inference and adaptability, albeit at significant computational expense.

Recent advancements (Shi et al., 2024; Zhang et al., 2023) improve model explainability and computational efficiency. In-context learning approaches (Lee et al., 2023; Ding et al., 2024) reformulate prediction tasks as sequence generation, optimizing historical data utilization. GenTKG (Liao et al., 2024) leverages the the few-shot tuning to expedite LLM reasoning, facilitating cross-domain generalization. The ONSEP framework (Yu et al., 2024) synergizes LLMs with TKGs for adaptive event prediction in dynamic settings, while CoH (Xia et al., 2024a) incorporates higher-order historical information as a modular enhancement, boosting graph model efficacy.

8 Conclusion

This paper introduces a training-free Analogical Replay (AnRe) reasoning framework for TKGF tasks. Through semantic-driven clustering and a dual history extraction module that combines long-term and short-term history, AnRe constructs comprehensive historical contexts for query events. It leverages LLMs to generate analogical reasoning examples as contextual input, enabling the model to deeply understand the historical evolution patterns of similar events. Experimental results demonstrate that AnRe significantly outperforms traditional trained models and existing LLM-based methods on four TKG benchmark datasets, validating its effectiveness in event prediction tasks. Furthermore, ablation studies confirm the critical role of the dual history extraction and analogical replay mechanisms. Future work will extend AnRe to other temporal reasoning tasks and optimize efficiency for larger knowledge graphs.

Limitations

Our method, due to its step-by-step nature, involves multiple calls to the LLM, leading to increased computational overhead and reasoning complexity. To fully utilize semantic information, we convert graph-structured information into textual form, which significantly expands the context input. As a result, the effectiveness of our method may be influenced by the length of the context. Additionally, the predictive accuracy of our framework relies on the model's precise analysis of analogical examples, which could pose significant challenges to the interpretability of the method. Currently, we employ only basic k-means clustering and BERT embeddings in the semantic-driven module, leaving substantial room for improvement in the depth of semantic information extraction. Moreover, we depend solely on the LLM's semantic understanding capabilities for extracting historical contexts, which still impose limitations on the comprehensiveness and accuracy of historical information. In future work, our approach could integrate graph-structured models with lower training costs to provide more accurate contextual backgrounds.

Acknowledgements

The research in this article is supported by the National Science Foundation of China (U22B2059, 62276083). We also appreciate the support from China Mobile Group Heilongjiang Co., Ltd. @ on our research, the research is jointly completed by both parties.

References

01. AI, :, Alex Young, Bei Chen, Chao Li, Chengen Huang, Ge Zhang, Guanwei Zhang, Heng Li, Jiangcheng Zhu, Jianqun Chen, Jing Chang, Kaidong Yu, Peng Liu, Qiang Liu, Shawn Yue, Senbin Yang, Shiming Yang, Tao Yu, Wen Xie, Wenhao Huang, Xiaohui Hu, Xiaoyi Ren, Xinyao Niu, Pengcheng Nie, Yuchi Xu, Yudong Liu, Yue Wang, Yuxuan Cai, Zhenyu Gu, Zhiyuan Liu, and Zonghong Dai. 2024. Yi: Open foundation models by 01.ai. *Preprint*, arXiv:2403.04652.

AI@Meta. 2024. Llama 3 model card.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Yuxiang Cai, Qiao Liu, Yanglei Gan, Changlin Li, Xueyi Liu, Run Lin, Da Luo, and JiayeYang JiayeYang. 2024a. Predicting the unpredictable: Uncertainty-aware reasoning over temporal knowledge graphs via diffusion process. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 5766–5778.

Zheng Cai, Maosong Cao, Haojiong Chen, Kai Chen, Keyu Chen, Xin Chen, Xun Chen, Zehui Chen, Zhi Chen, Pei Chu, Xiaoyi Dong, Haodong Duan, Qi Fan, Zhaoye Fei, Yang Gao, Jiaye Ge, Chenya Gu, Yuzhe Gu, Tao Gui, Aijia Guo, Qipeng Guo, Conghui He, Yingfan Hu, Ting Huang, Tao Jiang, Penglong Jiao, Zhenjiang Jin, Zhikai Lei, Jiaxing Li, Jingwen Li, Linyang Li, Shuaibin Li, Wei Li, Yining Li, Hongwei Liu, Jiangning Liu, Jiawei Hong, Kaiwen Liu, Kuikun Liu, Xiaoran Liu, Chengqi Lv, Haijun Lv, Kai Lv, Li Ma, Runyuan Ma, Zerun Ma, Wenchang Ning, Linke Ouyang, Jiantao Qiu, Yuan Qu, Fukai Shang, Yunfan Shao, Demin Song, Zifan Song, Zhihao Sui, Peng Sun, Yu Sun, Huanze Tang, Bin Wang, Guoteng Wang, Jiaqi Wang, Jiayu Wang, Rui Wang, Yudong Wang, Ziyi Wang, Xingjian Wei, Qizhen Weng, Fan Wu, Yingtong Xiong, Chao Xu, Ruiliang Xu, Hang Yan, Yirong Yan, Xiaogui Yang, Haochen Ye, Huaiyuan Ying, Jia Yu, Jing Yu, Yuhang Zang, Chuyu Zhang, Li Zhang, Pan Zhang, Peng Zhang, Ruijie Zhang, Shuo Zhang, Songyang Zhang, Wenjian Zhang, Wenwei Zhang, Xingcheng Zhang, Xinyue Zhang, Hui Zhao, Qian Zhao, Xiaomeng Zhao, Fengzhe Zhou, Zaida Zhou, Jingming Zhuo, Yicheng Zou, Xipeng Qiu, Yu Qiao, and Dahua Lin. 2024b. Internlm2 technical report. *Preprint*, arXiv:2403.17297.

Ziyang Chen, Jinzhi Liao, and Xiang Zhao. 2023. Multi-granularity temporal question answering over knowledge graphs.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Jacob Devlin. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Zifeng Ding, Heling Cai, Jingpei Wu, Yunpu Ma, Ruotong Liao, Bo Xiong, and Volker Tresp. 2024. zrllm: Zero-shot relational learning on temporal knowledge graphs with large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 1877–1895.

Alberto Garcia-Duran, Sebastijan Dumancic, and Mathias Niepert. 2018. Learning sequence encoders for temporal knowledge graph completion. *arXiv: Artificial Intelligence,arXiv: Artificial Intelligence*.

Alberto García-Durán, Sebastijan Dumančić, and Mathias Niepert. 2018. Learning sequence encoders

for temporal knowledge graph completion. *arXiv preprint arXiv:1809.03202*.

Julia Gastinger, Nils Steinert, Sabine Gründer-Fahrer, and Michael Martin. 2023. Dynamic representations of global crises: Creation and analysis of a temporal knowledge graph for conflicts, trade and value networks. In *D2R2*.

Zhen Han, Peng Chen, Yunpu Ma, and Volker Tresp. 2020a. Explainable subgraph reasoning for forecasting on temporal knowledge graphs. In *International conference on learning representations*.

Zhen Han, Peng Chen, Yunpu Ma, and Volker Tresp. 2020b. xerte: Explainable reasoning on temporal knowledge graphs for forecasting future links. *arXiv preprint arXiv:2012.15537*.

Zhen Han, Zifeng Ding, Yunpu Ma, Yujia Gu, and Volker Tresp. 2021. Learning neural ordinary equations for forecasting future links on temporal knowledge graphs. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, pages 8352–8364.

Zhen Han, Ruotong Liao, Beiyan Liu, Yao Zhang, Zifeng Ding, Jindong Gu, Heinz Koeppl, Hinrich Schuetze, and Volker Tresp. 2023. Enhanced temporal knowledge embeddings with contextualized language representations.

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. 2023a. Mistral 7b. *Preprint*, arXiv:2310.06825.

Jinhao Jiang, Kun Zhou, Zican Dong, Keming Ye, Wayne Xin Zhao, and Ji-Rong Wen. 2023b. Structgpt: A general framework for large language model to reason over structured data. *arXiv preprint arXiv:2305.09645*.

Woojeong Jin, Meng Qu, Xisen Jin, and Xiang Ren. 2019. Recurrent event network: Autoregressive structure inference over temporal knowledge graphs. *arXiv preprint arXiv:1904.05530*.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213.

Julien Leblay and Melisachew Wudage Chekol. 2018. Deriving validity time in knowledge graph. In *Companion of the The Web Conference 2018 on The Web Conference 2018 - WWW '18*.

Dong-Ho Lee, Kian Ahrabian, Woojeong Jin, Fred Morstatter, and Jay Pujara. 2023. Temporal knowledge graph forecasting without knowledge using in-context learning. *arXiv preprint arXiv:2305.10613*.

Kalev Leetaru and Philip A Schrodt. 2013. Gdelt: Global data on events, location, and tone, 1979–2012. In *ISA annual convention*, volume 2, pages 1–49. Citeseer.

Yujia Li, Shiliang Sun, and Jing Zhao. 2022. Tirgn: Time-guided recurrent graph network with local-global historical patterns for temporal knowledge graph reasoning. In *IJCAI*, pages 2152–2158.

Zixuan Li, Xiaolong Jin, Wei Li, Saiping Guan, Jiafeng Guo, Huawei Shen, Yuanzhuo Wang, and Xueqi Cheng. 2021. Temporal knowledge graph reasoning based on evolutional representation learning. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*, pages 408–417.

Ruotong Liao, Xu Jia, Yangzhe Li, Yunpu Ma, and Volker Tresp. 2024. Gentkg: Generative forecasting on temporal knowledge graph with large language models. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 4303–4317.

Zhenghao Liu, Chenyan Xiong, Maosong Sun, and Zhiyuan Liu. 2018. Entity-duet neural ranking: Understanding the role of knowledge graph semantics in neural information retrieval. *Cornell University - arXiv,Cornell University - arXiv*.

Stuart Lloyd. 1982. Least squares quantization in pcm. *IEEE transactions on information theory*, 28(2):129–137.

Matthew E Peters, Mark Neumann, Robert L Logan IV, Roy Schwartz, Vidur Joshi, Sameer Singh, and Noah A Smith. 2019. Knowledge enhanced contextual word representations. *arXiv preprint arXiv:1909.04164*.

Peter J Rousseeuw. 1987. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65.

Xiaoming Shi, Siqiao Xue, Kangrui Wang, Fan Zhou, James Zhang, Jun Zhou, Chenhao Tan, and Hongyuan Mei. 2024. Language models can improve event prediction by few-shot abductive reasoning. *Advances in Neural Information Processing Systems*, 36.

Haohai Sun, Jialun Zhong, Yunpu Ma, Zhen Han, and Kun He. 2021. Timetraveler: Reinforcement learning for temporal knowledge graph forecasting. *arXiv preprint arXiv:2109.04101*.

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

Qwen Team. 2024. Qwen2.5: A party of foundation models.

Rakshit Trivedi, Hanjun Dai, Yichen Wang, and Le Song. 2017. Know-evolve: Deep temporal reasoning for dynamic knowledge graphs. In *international conference on machine learning*, pages 3462–3471. PMLR.

Jiapu Wang, Kai Sun, Linhao Luo, Wei Wei, Yongli Hu, Alan Wee-Chung Liew, Shirui Pan, and Baocai Yin. 2024. Large language models-guided dynamic adaptation for temporal knowledge graph reasoning. *arXiv preprint arXiv:2405.14170*.

Xiang Wang, Xiangnan He, Yixin Cao, Meng Liu, and Tat-Seng Chua. 2019. Kgat: Knowledge graph attention network for recommendation. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 950–958.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Yuwei Xia, Ding Wang, Qiang Liu, Liang Wang, Shu Wu, and Xiao-Yu Zhang. 2024a. Chain-of-history reasoning for temporal knowledge graph forecasting. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 16144–16159.

Yuwei Xia, Mengqi Zhang, Qiang Liu, Liang Wang, Shu Wu, and Xiaoyu Zhang. 2024b. Metatkg++: Learning evolving factor enhanced meta-knowledge for temporal knowledge graph reasoning. *Pattern Recognition*, page 110629.

Sheng Xiang, Dawei Cheng, Chencheng Shang, Ying Zhang, and Yuqi Liang. 2022. Temporal and heterogeneous graph neural network for financial time series prediction.

Wenjie Xu, Ben Liu, Miao Peng, Xu Jia, and Min Peng. 2023a. Pre-trained language model with prompts for temporal knowledge graph completion. *arXiv preprint arXiv:2305.07912*.

Yi Xu, Junjie Ou, Hui Xu, and Luoyi Fu. 2023b. Temporal knowledge graph reasoning with historical contrastive learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 4765–4773.

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, 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, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zhihao Fan. 2024. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*.

Linyao Yang, Hongyang Chen, Zhao Li, Xiao Ding, and Xindong Wu. 2023. Chatgpt is not enough: Enhancing large language models with knowledge graphs for fact-aware language modeling. *arXiv preprint arXiv:2306.11489*.

Weihao Yu, Zihang Jiang, Yanfei Dong, and Jiashi Feng. 2020. Reclor: A reading comprehension dataset requiring logical reasoning. *arXiv preprint arXiv:2002.04326*.

Xuanqing Yu, Wangtao Sun, Jingwei Li, Kang Liu, Chengbao Liu, and Jie Tan. 2024. Onsep: A novel online neural-symbolic framework for event prediction based on large language model. *arXiv preprint arXiv:2408.07840*.

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

Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi Hu, Tianwei Zhang, Fei Wu, et al. 2023. Instruction tuning for large language models: A survey. *arXiv preprint arXiv:2308.10792*.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. 2023. A survey of large language models. *arXiv preprint arXiv:2303.18223*.

Cunchao Zhu, Muhao Chen, Changjun Fan, Guangquan Cheng, and Yan Zhang. 2021. Learning from history: Modeling temporal knowledge graphs with sequential copy-generation networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 4732–4740.

Algorithm 1 Analogical Replay for TKGF

Require: Entity Set, $\mathcal{V}$
Require: History Events Set at time t_n, $\mathcal{H}_n$
Require: Large Language Model, LLM
Require: Query, $q/(s_q, r_q, ?, t_{n+1})$
Require: Number of Analogical Examples, a
Ensure: Object Entity Prediction, o_q

$\mathcal{V}' \leftarrow \text{Clustering}(\mathcal{V})$
$\mathcal{X} \leftarrow \text{ClusterRetriever}(\mathcal{V}', s_q)$
$\mathcal{A}, \mathcal{P} \leftarrow \emptyset, \emptyset$
for $s_i \in \mathcal{X}$ **do**
  if $s_i \neq s_q$ **then**
    $H_i, e_i \leftarrow \text{CandiFilter}(\mathcal{H}_n, r_q, s_i)$
    $\mathbb{H}_i^S \leftarrow \text{S-Term}(H_i)$
    $\mathbb{H}_i^L \leftarrow \text{L-Term}(LLM(H_i, \text{Masked}(e_i)))$
    $\mathbb{H}_i \leftarrow \mathbb{H}_i^S \cup \mathbb{H}_i^L$
    $\mathcal{A} \leftarrow \text{PickTop}(\mathcal{A}, a, (\mathbb{H}_i, e_i))$
  else
    $H_q, O_q \leftarrow \text{CandiFilter}(\mathcal{H}_n, r_q, s_q)$
    $\mathbb{H}_q^S \leftarrow \text{S-Term}(H_q)$
    $\mathbb{H}_q^L \leftarrow \text{L-Term}(LLM(H_q, q))$
    $\mathbb{H}_q \leftarrow \mathbb{H}_q^S \cup \mathbb{H}_q^L$
  end if
end for
for $(\mathbb{H}_{a_i}, e_{a_i}) \in \mathcal{A}$ **do**
  $ex_{a_i} \leftarrow \text{Replay}(LLM(\mathbb{H}_{a_i}, e_{a_i}))$
  $\mathcal{P} \leftarrow \mathcal{P} \cup ex_{a_i}$
end for
$o_q \leftarrow \text{Infer}(LLM(\mathcal{P}, \mathbb{H}_q, q, O_q))$
return o_q

A Algorithm of AnRe

We provide the algorithm of our framework in Algorithm 1.

B Details of Models

InternLM2 (Cai et al., 2024b) is a model fine-tuned on the internlm2-base architecture using specialized domain corpora, capable of handling 32K context inputs. We employ both the 7B and 20B versions of InternLM2 in our experiments.

Qwen2.5 (Yang et al., 2024; Team, 2024) is the latest series of Qwen large language models, featuring Long-context Support up to 128K tokens.We employ the *Qwen2.5-7B-Instruct* version for our experiments.

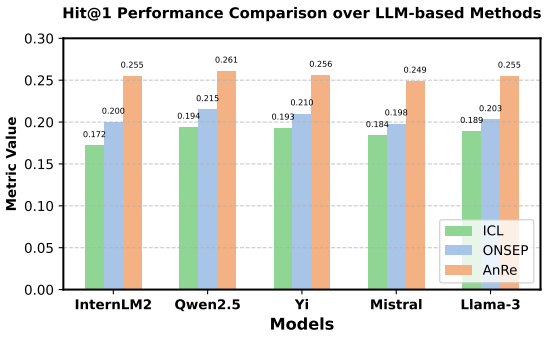

Figure 7: The performance comparison of ICL, ONSEP, and AnRe using different LLMs, with the results selected based on the Hit@1 metric on the ICEWS18 dataset.

Yi (AI et al., 2024) We access the *Yi-6B-200K* version, supporting up to 200K context inputs.

Mistral (Jiang et al., 2023a) We utilize the *Mistral-7B-Instruct-v0.3* model, supporting a context length of 32K tokens.

Llama-3 (AI@Meta, 2024) We access the *Llama-3-8B-Instruct-262k* version for our experiments, which is capable of handling 262K context inputs.

C Analysis of Model Scales

As shown in Table 4, the performance of the 20B model demonstrated slight improvements across all metrics compared to the 7B model. Notably, under the ICEWS14 and ICEWS18 datasets, the Hit@1 metric increases by 1.45% and 1.96%, respectively. This indicates that models with larger parameter sizes can better comprehend longer contextual inputs and capture historical development patterns within analogical event chains. However, compared to the effects of ablation experiments and hyperparameter adjustments, the impact of model scale on predictive performance is relatively minor. This observation underscores the efficiency of our method when applied to smaller-scale models.

D Analysis of Context Length Limitations

As demonstrated in Table 6, the inclusion of a single analogical example yields optimal results across all metrics, achieving a 15.9% relative improvement in Hit@1 compared to zero-shot prompting. However, doubling the examples to two not only fails to produce significant performance gains but also incurs substantial computational

Model	ICEWS14			ICEWS05-15			ICEWS18		
	Hit@1	Hit@3	Hit@10	Hit@1	Hit@3	Hit@10	Hit@1	Hit@3	Hit@10
InternLM2-7B	0.346	0.470	0.608	0.389	0.551	0.678	0.255	0.371	0.554
InternLM2-20B	0.351	0.472	0.610	0.391	0.554	0.681	0.260	0.372	0.555
Δ *Improve*	1.45%	0.43%	0.33%	0.51%	0.54%	0.44%	1.96%	0.27%	0.18%

Table 4: The performance with different LLM parameters scales.

costs—input tokens increase by 183% and processing time by 285%. We attribute this phenomenon to two factors: (1) the introduction of noisy or redundant contextual information with additional examples, which may interfere with the model's reasoning focus, (2) the inherent constraints of the LLM's token capacity, where longer contexts disproportionately increase memory and time overhead without commensurate accuracy benefits.

E Prompt Templates

The prompt templates in our framework will be presented from Table 7 to 9.

F Reasoning Demonstrations

In Figure 8 to 10, we provide demonstrations of the reasoning process using LLMs across different modules.

G Formal Definition of Notations

We describe the formal definition of the notation used in the pseudocode algorithm and the formulas in this paper, as shown in Table 10.

LLM	Methods	ICEWS14			ICEWS05-15			ICEWS18		
		Hit@1	Hit@3	Hit@10	Hit@1	Hit@3	Hit@10	Hit@1	Hit@3	Hit@10
InternLM2	ICL	0.301	0.432	0.560	0.353	0.507	0.647	0.172	0.289	0.434
	ONSEP	0.330	0.464	0.570	0.386	0.546	0.662	0.200	0.324	0.443
	AnRe	0.346	0.470	**0.608**	0.389	0.551	0.678	0.255	0.371	0.554
Qwen2.5	ICL	0.308	0.437	0.565	0.360	0.510	0.649	0.194	0.294	0.437
	ONSEP	0.345	0.467	0.578	0.388	0.547	0.666	0.215	0.336	0.484
	AnRe	**0.351**	0.468	**0.608**	**0.390**	**0.553**	0.677	**0.261**	**0.372**	**0.556**
Yi	ICL	0.305	0.433	0.568	0.355	0.509	0.648	0.193	0.291	0.436
	ONSEP	0.341	0.464	0.591	0.387	0.544	0.670	0.210	0.327	0.478
	AnRe	0.347	**0.471**	0.607	0.387	0.550	**0.679**	0.256	0.369	0.547
Mistral	ICL	0.306	0.433	0.566	0.352	0.504	0.644	0.184	0.288	0.434
	ONSEP	0.342	0.465	0.571	0.385	0.545	0.654	0.198	0.328	0.445
	AnRe	0.343	0.467	0.598	0.386	0.542	0.660	0.249	0.367	0.544
Llama-3	ICL	0.305	0.434	0.563	0.353	0.506	0.648	0.189	0.288	0.436
	ONSEP	0.345	0.469	0.581	0.387	0.546	0.663	0.203	0.329	0.454
	AnRe	0.348	0.469	0.604	**0.390**	0.547	0.671	0.255	0.368	0.547

Table 5: The performance comparison of ICL, ONSEP, and AnRe using different LLMs. The best results within the same metric are highlighted in **bold**, while the best results within the same LLM are underlined.

Demonstration of Historical Event Selection

There is a question and some historical events.Please select the historical event that is most helpful in answering this question from these historical events and return the number before the event.
Question:*On 16 October 2018, Manohar Parrikar expressed intent to meet or negotiate with whom?*
Historical events:
1.*On 14 June 2018, Manohar Parrikar expressed intent to meet or negotiate with India.*
2.*On 14 June 2018, the Director General (India) hosted a visit for Manohar Parrikar.*
3.*On 14 June 2018, the Director General (India) praised or endorsed Manohar Parrikar.*
4.*On 14 June 2018, Manohar Parrikar made a visit to the Director General (India).*
5.*On 14 June 2018, Manohar Parrikar made a visit to the Health Ministry of India.*
6.*On 14 June 2018, the Health Ministry (India) hosted a visit by Manohar Parrikar.*
7.*On 14 June 2018, the Health Ministry (India) praised or endorsed Manohar Parrikar.*
8.*On 14 June 2018, Bharatiya Janata made a statement about Manohar Parrikar.*
9.*On 14 June 2018, India hosted a visit by Manohar Parrikar.*
10.*On 14 June 2018, Manohar Parrikar made a visit to India.*
...
Your choice is:

Figure 8: A demonstration of historical event selection in PDC.

Demonstration of Analysis Process Construction

Here are some time-ordered historical event chains and a question-answer pair. Please analyze how the answer to the question is derived based on the historical development process in the event chain.
Historical Events:
On 24 March 2018, Leo Varadkar praised or endorsed the United Kingdom.
On 26 March 2018, Leo Varadkar made a statement about Sean Treacy.
...
On 12 June 2018, Leo Varadkar made a statement about the Government of the United Kingdom.
On 16 June 2018, Micheál Martin made a statement about Leo Varadkar.
...
On 25 July 2018, Leo Varadkar demanded a meeting for negotiation with Romania.
On 26 July 2018, Leo Varadkar appealed for a change in leadership involving Micheál Martin.
...
Question:*On 13 October 2018, Leo Varadkar expressed intent to meet or negotiate with whom?*
Answer:*Micheál Martin*
Your Analysis:

Analysis:
The timeline shows a series of interactions between Leo Varadkar and Micheál Martin, starting with statements and escalating to appeals and rejections.
On 21 July 2018, Leo Varadkar directly appealed to Micheál Martin, indicating a desire to engage in dialogue or negotiation.
The rejection by Micheál Martin on 5 September 2018 suggests that the relationship was strained, but it also underscores the importance of Micheál Martin as a key figure in Leo Varadkar's political landscape.
By 13 October 2018, the context of these interactions suggests that Leo Varadkar's intent to meet or negotiate would logically be directed towards Micheál Martin, given their history of engagement and the unresolved issues between them.
...

Figure 9: A demonstration of analysis process construction in Analogical Replay.The shaded portion represents the analysis process provided by the LLM.

Demonstration of Object Entity Prediction

You will be provided with some historical events in chronological order, a question, and several candidate answers. Based on the development of these historical events, please deduce the most likely correct answer entity for the question and output its corresponding number. Below is one reasoning example.
Analogical Example 1:
Historical Events:
On 24 March 2018, Leo Varadkar praised or endorsed the United Kingdom.
...
On 25 July 2018, Leo Varadkar demanded a meeting for negotiation with Romania.
On 26 July 2018, Leo Varadkar appealed for a change in leadership involving Micheál Martin.
...
Question:On 13 October 2018, Leo Varadkar expressed intent to meet or negotiate with whom?
Answer:Micheál Martin.The timeline shows a series of interactions between Leo Varadkar and Micheál Martin, starting with statements and escalating to appeals and rejections. ...

Please learn from this example and then provide the most likely correct answer number for the question.
Historical Events:
On 14 June 2018, the Director General (India) praised or endorsed Manohar Parrikar.
...
On 15 October 2018, the Head of Government (India) made a statement about Manohar Parrikar.
On 15 October 2018, Manohar Parrikar expressed intent to meet or negotiate with India.
Question:*On 16 October 2018, Manohar Parrikar expressed intent to meet or negotiate with whom?*
Candidate Answers:
1.*Amit Shah*
2.*Intelligence (India)*
3.*Businessperson (India)*
4.*Non-Governmental Organizations*
...
Your choice is:

Figure 10: A demonstration of object entity prediction in Analogical Replay.(1 example)

# Example	Hit@1	Hit@3	Hit@10	Token	Time
0	0.220	0.364	0.513	2687.43	39.28s/it
1	**0.255**	**0.371**	**0.554**	5102.67	63.82s/it
2	0.238	0.365	0.518	**7617.34**	**112.15s/it**

Table 6: Performance comparison with varying numbers of analogical examples.

Prompt Template in PDC

There is a question and some historical events. Please select the historical event that is most helpful in answering this question from these historical events and return the number before the event.
Question:{*query*}
Historical events:{*label:event*}
Your choice is:

Table 7: Prompt template for historical event selection in PDC.

Prompt Template for APC

Here are some time-ordered historical event chains and a question-answer pair. Please analyze how the answer to the question is derived based on the historical development process in the event chain.
Historical Events: $\{\mathbb{H}_{a_i}\}$
Question: $\{Masked(e_{a_i})\}$
Answer: $\{o_{a_i}\}$
Your Analysis:

Table 8: Prompt template for analysis process construction in Analogical Replay.

Prompt Template for OEP

You will be provided with some historical events in chronological order, a question, and several candidate answers. Based on the development of these historical events, please deduce the most likely correct answer entity for the question and output its corresponding number. Below are a reasoning examples.
...
Analogical Example ex_{a_i}:
Historical Events:$\{\mathbb{H}_{a_i}\}$
Question: $\{Masked(e_{a_i})\}$
Answer: $\{o_{a_i}, p_{a_i}\}$
...
Please learn from these examples and then provide the most likely correct answer number for the question.
Historical Events:$\{\mathbb{H}_q\}$
Question: $\{q\}$
Candidate Answers: $\{O_q\}$
Your choice is:

Table 9: Prompt template for object entity prediction in Analogical Replay.

Symbol	Description
(i) Semantic-driven Historical Clustering	
$q/(s_q, r_q, ?, t_{n+1})$	Target query, where s_q is the subject in the query, r_q is the relation, and t_{n+1} is the timestamp indicating when the event occurs.
$\mathcal{X}$	The set of entities obtained after clustering based on semantic similarity, where all entities in the set are semantically similar to s_q.
$\mathcal{V}, \mathcal{V}'$	The set of all entities and the set of sets of similar entities, respectively.
k	The size of $\mathcal{V}'$, which represents the number of clusters formed by grouping all entities based on semantic similarity.
$\mathcal{H}_n$	The set of all historical events that occurred before t_{n+1}.
s_i	Any similar entity in $\mathcal{X}$ other than s_q.
H_i/H_q	The set of events in which s_i/s_q appears, also referred to as the set of related events of s_i/s_q.
e_i	The event involving s_i that is most similar to the target query.
O_q	The set of candidate answer entities for the target query.
CandiFilter(...)	The Candidate History Filter module, which filters out the corresponding H_i and e_i for s_i, as well as the corresponding H_q and O_q for s_q.
(ii) Dual History Extraction	
S-Term(...), L-Term(...)	The short-term history retriever and the long-term history retriever, respectively.
$\mathbb{H}_i^S/\mathbb{H}_q^S$	The set of short-term histories corresponding to s_i/s_q.
l	Hyperparameter: the length of short-term history.
q_i/Masked(e_i)	A similar query q_i for q,constructed by masking the tail entity in e_i.
$LLM(...)$	The large language model receives structured prompts and outputs example analyses or probability distributions of numerical mappings.
$H_i^{t_j}/H_q^{t_j}$	The set of relevant histories for s_i/s_q at timestamp t_j.
$\mathbb{H}_i^L/\mathbb{H}_q^L$	The set of long-term histories corresponding to s_i/s_q.
α	Hyperparameter: controls the rate at which the dynamic threshold changes over time differences.
$\mathbb{H}_i/\mathbb{H}_q$	The set of combined long-term and short-term historical events corresponding to s_i/s_q.
(iii) Analogical Replay	
$\mathcal{A}$	The set of tuples consisting of analogical events and their corresponding histories.
PickTop(...)	Based on the semantic similarity between similar events and the target query, the top a events are selected as analogical events for constructing analogical reasoning examples.
a	The number of analogical reasoning examples.
ex_{a_i}	The analogical reasoning example, consisting of three parts: the analogical event, its corresponding history, and the model's analysis.
$\mathcal{P}$	The set of analogical reasoning examples.
o_q	The predicted tail entity for the target query.

Table 10: The formal definition of notations used in the algorithms and formulas within this paper.