An Adaptive Logical Rule Embedding Model for Inductive Reasoning over Temporal Knowledge Graphs

Xin Mei[*], **Libin Yang**[*], **Zuowei Jiang**, **Xiaoyan Cai**[†]
Northwestern Polytechnical University, Xi'an, China
meixin@mail.nwpu.edu.cn, libiny@nwpu.edu.cn
jiangzw@mail.nwpu.edu.cn, xiaoyanc@nwpu.edu.cn

Abstract

Temporal knowledge graphs (TKGs) extrapolation reasoning predicts future events based on historical information, which has great research significance and broad application value. Existing methods can be divided into embedding-based methods and logical rule-based methods. Embedding-based methods rely on learned entity and relation embeddings to make predictions and thus lack interpretability. Logical rule-based methods bring scalability problems due to being limited by the learned logical rules. We combine the two methods to capture deep causal logic by learning rule embeddings, and propose an interpretable model for temporal knowledge graph reasoning called **a**daptive **l**ogical **r**ule **e**mbedding model for **i**nductive **r**easoning (ALRE-IR). ALRE-IR can adaptively extract and assess reasons contained in historical events, and make predictions based on causal logic. Furthermore, we propose a one-class augmented matching loss for optimization. When evaluated on ICEWS14, ICEWS0515 and ICEWS18 datasets, the performance of ALRE-IR outperforms other state-of-the-art baselines. The results also demonstrate that ALRE-IR still shows outstanding performance when transferred to related dataset with common relation vocabulary, indicating our proposed model has good zero-shot reasoning ability.[1]

1 Introduction

Knowledge graphs (KGs) are a form of structured human knowledge. They represent events as triples (subject, relation, object), where subject and object are entities. Entities usually are objects and abstract concepts in the real world, and relations represent relationships between entities. KGs have caused great research both in academia and industry (Dong et al., 2014; Nickel et al., 2015; Wang et al., 2017; Hogan et al., 2021), and have been widely used in many real-world applications including relation extraction (Min et al., 2013; Zeng et al., 2015), entity linking (Hua et al., 2015; Mendes et al., 2011), and question answering (Luo et al., 2018; Yih et al., 2015). However, most of the knowledge graphs are incomplete (Shi and Weninger, 2018; Toutanova and Chen, 2015), which affects their effectiveness and limits the performance of KG-based applications. Reasoning over KGs aims to infer new conclusions based on existing data and predict the missing events, which can effectively alleviate this problem. Traditional knowledge graphs contain only static events, and there is a large amount of available event data with temporal correlations, where entities interact differently over time. Therefore, many temporal knowledge graphs (TKGs) composed of entity interaction data with temporal attributes have emerged (Boschee et al., 2015; Gottschalk and Demidova, 2018, 2019).TKGs extend static triples with timestamp to represent dynamic events in the form of quadruples (subject, relation, object, timestamp), where timestamp represents valid time of static triple. Compared with traditional static KGs, TKGs have complex temporal dynamic characteristics, which increase difficulty of reasoning on TKGs.

Reasoning over a TKG primarily has two settings, interpolation (Goel et al., 2020) and extrapolation (Trivedi et al., 2017). Given events within time interval $[t_0, t_T]$, interpolation attempts to infer missing events that happened in $[t_0, t_T]$, while extrapolation predicts future missing events for time $t > t_T$. Extrapolation reasoning learns hidden connections between events from observed historical KGs and then predicts new events at future timestamps (Korkmaz et al., 2015; Muthiah et al., 2015; Phillips et al., 2017), which can be applied in practical scenarios such as disaster relief (Signorini et al., 2011) and financial analysis (Bollen

[*] Equal contribution
[†] Corresponding author
[1] Our code is released at https://github.com/mxadorable/ALRET-IR.

Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 7304–7316
December 7-11, 2022

et al., 2011).This paper focuses on extrapolation reasoning task.

Recently, many research efforts have been put into extrapolation reasoning over TKGs and realize excellent prediction performance (Tao et al., 2021). These methods can be divided into two categories: embedding-based methods and logical rule-based symbolic methods. Embedding-based methods such as RE-Net (Jin et al., 2020), CyGNet (Zhu et al., 2021), TIE (Wu et al., 2021) and RE-GCN (Li et al., 2021) can capture complex information in TKG, but the black-box property of embeddings make them lack interpretability and are not suitable for many practical applications. Some researchers propose to create logical rules for reasoning to enhance credibility and utility, such as Streamlearner (Omran et al., 2019) and Tlogic (Liu et al., 2022). They employ statistical-based measures to assess confidence of rules and make predictions based on learned rules. However, the learned rules are limited, which makes the model have scalability problems and is not suitable for large-scale datasets in reality.

To alleviate the above problems, we propose an adaptive logical rule embedding model for inductive reasoning (ALRE-IR) on temporal knowledge graphs. It can effectively capture deep structure of TKG and mine potential logical rules. Logical rules are represented by a sequence of relations. Therefore, relations are the core features we focus on when mining rules, and entities are just tools for extracting relation paths. First, we extract relation paths from historical subgraphs and learn embeddings of relation paths that contain historical semantics. We then match these relation paths with current events to obtain rules and assess confidence of the rules based on interpretable causal logic. Finally, the quadruple can be scored according to confidence of the rules. We design training tasks from a coarse-grained quadruple perspective and a fine-grained rule perspective, respectively, and propose a one-class augmented matching loss to optimize our proposed adaptive logical rule embedding model. During the inference process, our model can adaptively extract and learn relation path features based on historical information, assess the confidence of corresponding rules, and predict missing entities. Furthermore, our trained model can be applied to new datasets with a common relation vocabulary for zero-shot reasoning.

In summary, this paper makes the following four contributions:

(1) An interpretable temporal knowledge graph reasoning method is developed, which can perform effective inductive reasoning.

(2) An adaptive logical rule embedding model is proposed, which can autonomously extract and assess rules based on historical features.

(3) A one-class augmented matching loss is designed to train the model from a coarse-grained quadruple perspective and a fine-grained rule perspective, respectively.

(4) Thorough experimental studies are conducted, and experimental results show that our proposed ALRE-IR model outperforms state-of-the-art baselines.

2 Related Work

2.1 Static Knowledge Graph (KG) Reasoning

Common static KG reasoning models mainly focus on knowledge representation learning, that is, learning low-dimensional vector representations of entities and relations. These models are mainly divided into three categories: translation based models, semantic matching based models, and neural network based models. Translation based models regard the relation as a translation vector from a subject entity to an object entity, such as TransE (Bordes et al., 2013), TransH (Wang et al., 2014), and TransR (Lin et al., 2015). Semantic matching based models (e.g. ComplEx (Trouillon et al., 2016), DistMult (Yang et al., 2015) and RotatE (Sun et al., 2019)), assume that the score of a triple can be factorized into several tensors, and use triangular norm to measure the rationality of facts. Neural network based models use deep neural networks to learn network embeddings. For example, ConvE (Dettmers et al., 2018) and ConvKB (Nguyen et al., 2018) use convolutional neural networks to learn interactions between entities and relation. In addition, some models utilize graph neural networks which have outstanding performance in graph representation learning to embed KG, such as R-GCN (Schlichtkrull et al., 2018), A2N (Bansal et al., 2019), and RGHAT (Zhang et al., 2020).

2.2 Temporal Knowledge Graph (TKG) Reasoning

TKG reasoning can have two settings: extrapolation reasoning and interpolation reasoning. For interpolation reasoning, researchers complete missing events in past timestamps by adding temporal

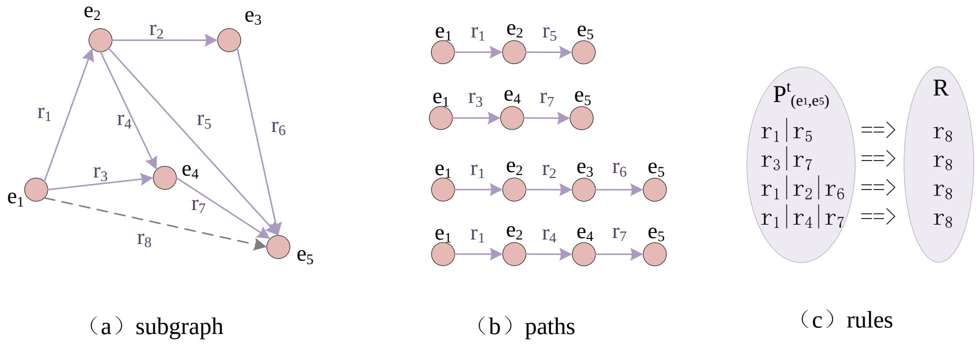

Figure 1: Rule extraction based on relation paths.

information to the static KG representation learning method. TTransE (Leblay and Chekol, 2018) is an extension of TransE, which embeds temporal information into score function. HyTE (Dasgupta et al., 2018) improves TransH by replacing the unit normal vector of the hyperplane projection with the normal vector related to timestamp. TA-DistMult (Garcia-Duran et al., 2018) uses recurrent neural networks to embed time into relation embeddings.

Unlike interpolation reasoning, extrapolation reasoning predicts new events in the future based on historical facts. Existing methods for extrapolating reasoning can be divided into embedding-based methods and logical rule-based methods. Embedding-based methods include RE-NET (Jin et al., 2020), CyGNet (Zhu et al., 2021), RE-GCN (Li et al., 2021) and xERTE (Han et al., 2021). They capture temporal information either by learning embeddings for each timestamp, or learning evolutional embeddings of entities and relations over time. Although these methods can capture complex features, they rely on trained embeddings and cannot make inductive predictions for events containing new entities, relations, or timestamps. Recently, some researchers have proposed logical rule-based interpretable methods, such as AnyBURL(Meilicke et al., 2020), StreamLearner (Omran et al., 2019) and TLogic (Liu et al., 2022). These methods mine logical rules from datasets through random walks, and design measures to assess confidence of candidate logical rules. Therefore, quality of the logical rules they learn depends largely on the measure chosen. Furthermore, these methods apply learned rules for reasoning and cannot adapt to new patterns of logical rules.

3 Preliminaries

Temporal Knowledge Graph (TKG). A TKG consists of dynamic events, an event is represented in the form of a quadruple (s, r, o, t) consisting of a subject entity $s \in E$, a relation $r \in \Upsilon$, an object entity $o \in E$ and a timestamp $t \in \Gamma$. E and Υ denote entity set and relation set respectively, and Γ represents the set of timestamps.

Link Prediction. Given a missing temporal quadruple (event), link prediction aims to infer the missing part, such as predicting object entity given $(s, r, ?, t)$ or predicting subject entity given $(?, r, o, t)$ or predicting relation given $(s, ?, o, t)$. For each quadruple, the training objective function is optimized to make the correct quadruple score higher than the incorrect quadruple, it is generally defined as a score function $g(s, r, o, t) \in R$.

Temporal logical rules. We predict future events by mining logical causal relationships between events. As shown in Figure 1, we take an event (e_1, r_8, e_5, t) as an example, and mine temporal logical rules contained in it according to the historical information in the previous m timestamps. Figure 1(a) represents a subgraph composed of events that occurred in the previous m timestamps. Based on this subgraph, we mine all rules that might lead to the current event. Figure 1(b) shows all paths from e_1 to e_5, we can extract relations to get four possible logical rules in Figure 1(c). Each rule $R^t(p, r)$ consists of a path $p \in P^t_{(s,o)}$ and a relation $r \in \Upsilon$, p represents the historical reason and r represents the result at the current timestamp.

4 Method

We propose an interpretable model for temporal knowledge graph reasoning called adaptive logi-

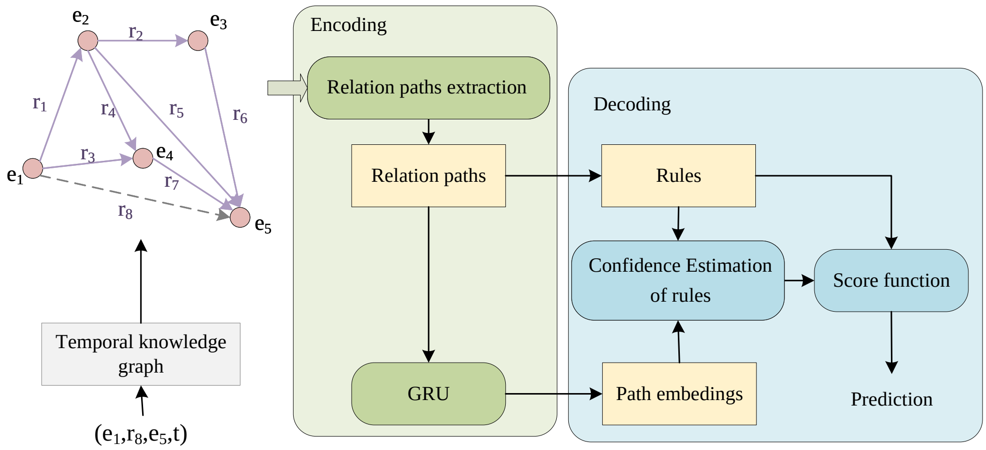

Figure 2: Architecture of the proposed model.

cal rule embedding model for inductive reasoning (ALRE-IR). It uses relation paths to represent logical rules implicit in the knowledge graph, and captures complex semantic features in the knowledge graph by learning embeddings of relation paths. It can adaptively extract possible rules, learn rules' embeddings, score rules based on interpretable causal logic, and finally make predictions based on relative confidence of rules. Our inductive reasoning model is composed of three parts as follows:

Encoding, which walks out all historical relation paths for each input quadruple, learns embeddings of all relation paths.

Decoding, which scores quadruples according to all the temporal logical rules associated with them.

Training, which proposes a one-class augmented matching loss to optimize the model, so that the model can adaptively learn reasonable logical rules.

The overall architecture of the model is shown in Figure 2, details of the model will be elaborated as follows.

4.1 Encoding

Existing representation learning based temporal knowledge reasoning methods make predictions by learning evolutional embeddings of entities. We propose to infer missing links according to causal logical rules, and encode historical information to find the reason that leads to the event. In this part, we mine relation logical rules contained in historical subgraphs, and learn embeddings of relation paths.

4.1.1 Relation Paths Extraction

We take entities as nodes and relations as edges, and construct a relation graph according to all events from timestamp $t-m$ to $t-1$. For events (s, r, o, t), we take the k-hop neighbors around node s and node o respectively to get two subgraphs, and take the intersection of the two subgraphs. Then we remove independent nodes and nodes whose distance from node s or node o is greater than k. By doing this, we can obtain a subgraph containing all paths between node s and node o whose length does not exceed $k+1$. After that, we extract all paths between node s and node o on the subgraph, and remove entities to get relation paths.

4.1.2 Relation Paths Embedding

We learn relation path sequence embeddings to capture logical semantics implied in the relation paths, which reflect spatial logical correlation of the two entities.

In our approach, we exploit Gated Recurrent Unit (GRU), a popular variant of RNNs, to capture features of relation paths. Gated recurrent neural networks (Gated RNNs) have been successfully applied in processing data with sequence characteristics, i.e., data that conform to temporal, logical, or other orderings. RNNs can capture relationships between sequential data, and mine sequential information and semantic information in the data. The reason why RNNs can solve the sequence problem is that it can remember information at each moment. The hidden layer at each moment is not only determined by the input layer at this moment, but also by the output of the hidden layer at the previous moment. A simple recurrent unit can be represented as:

$$\mathbf{h}_t = f(\mathbf{W} \cdot \mathbf{x}_t + \mathbf{U} \cdot \mathbf{h}_{t-1} + \mathbf{b}) \quad (1)$$

where $\mathbf{x}_t$ is the input vector at timestamp t, $\mathbf{h}_t$ is the hidden states at timestamp t, $\mathbf{W}$ and $\mathbf{U}$ are two trainable weight parameters, and $f(\cdot)$ is an activation function.

The relation paths we extract are of variable length, and GRU can easily process such sequences and get embeddings for each path. GRU controls the flow of information through two learnable gates, called update gate and reset gate. The update gate controls historical memory information to be retained, and the reset gate controls information to be forgotten. The specific formula of GRU model is as follows:

$$\mathbf{z}_t = \sigma(\mathbf{W}_z \cdot \mathbf{x}_t + \mathbf{U}_z \cdot \mathbf{h}_{t-1} + \mathbf{b}_z) \quad (2)$$

$$\mathbf{r}_t = \sigma(\mathbf{W}_r \cdot \mathbf{x}_t + \mathbf{U}_r \cdot \mathbf{h}_{t-1} + \mathbf{b}_r) \quad (3)$$

$$\tilde{\mathbf{h}_t} = tanh(\mathbf{W} \cdot \mathbf{x}_t + \mathbf{U} \cdot (\mathbf{r}_t \odot \mathbf{h}_{t-1}) + \mathbf{b}) \quad (4)$$

$$\mathbf{h}_t = \mathbf{z}_r \odot \tilde{\mathbf{h}_t} + (1 - \mathbf{z}_r) \odot \mathbf{h}_{t-1} \quad (5)$$

where $\mathbf{z}_t$ is the update gate and $\mathbf{r}_t$ is the reset gate. $\mathbf{h}_{(t-1)}$ represents the hidden state at timestamp $t-1$, which acts as the neural network memory, containing information of the previous input. σ is the sigmoid function.

4.2 Decoding

4.2.1 Confidence Estimation of rules

Relation paths extracted based on events (s, r, o, t) represent possible "reasons" of events. According to the "result" r, we find reasonable ones from these reasons, i.e. find possible matching rules.

For an event (s, r, o, t), we get all relation paths between the subject entity s and object entity o, and the corresponding embeddings $\mathbf{P}^t_{(s,o)}$. Taking the path $p_i \in P^t_{(s,o)}$ as "reason" and relation r as "result", a rule $R^t(p, r)$ is obtained. Then, we estimate the rule confidence by capturing the interaction between path p_i and relation r. We define two confidence estimation functions as:

Similarity matching:

$$f(p_i, r) = cos(\mathbf{p}_i, \mathbf{r}) \quad (6)$$

where cos represents cosine similarity. This function measures interaction score of path and relation using cosine similarity.

Concatenation combination:

$$f(p_i, r) = \sigma(\mathbf{W}(\mathbf{p}_i \| \mathbf{r})) \quad (7)$$

where σ is the sigmoid function, $\|$ represents concatenation operation. This function concatenates path and relation embeddings, capturing their interaction score with a linear projection, and $\mathbf{W}$ represents a projection matrix.

4.2.2 Score function

We predict the missing quadruple $(s, r, ?, t)$ based on the learned path embeddings. For a candidate target entity o, the corresponding quadruple is (s, r, o, t). The score function is defined as:

$$g(s, r, o, t) = max_{\mathbf{p}_i \in \mathbf{P}^t_{(s,o)}} f(\mathbf{p}_i, r) \quad (8)$$

Taking entity s as the starting node and entity o as the ending node, we extract relation paths that represent all possible reasons of the current event. Combine these paths with relation r to generate a set of rules. The rule with the highest confidence indicates that the relation path in this rule is the most reasonable "reason" for the current quadruple, and we use confidence of this rule as the score of the quadruple.

4.3 Training

In this subsection we introduce an objective function for training. For relation r, we train the model to find matching paths in historical events (correct rules). However, the biggest difficulty is that we do not know which path-relation pair is matched, i.e., there is no correct rule for training. To this end, we design training tasks from a coarse-grained quadruple perspective and a fine-grained rule perspective, respectively, and propose a one-class augmented matching loss.

4.3.1 Training from quadruple perspective

From the quadruple perspective, similar to embedding-based methods, we design a main loss function according to the quadruple score, in order to let the correct quadruple score higher and the wrong quadruple score lower. The loss function is soft-margin loss as:

$$L_1 = \sum_{(s,r,o,t) \in Q \bigcup Q'} log(1 + exp(l \cdot g(s, r, o, t))) \quad (9)$$

$$l = \begin{cases} 1, & (s, r, o, t) \in Q \\ -1, & (s, r, o, t) \in Q' \end{cases} \quad (10)$$

where Q is the set of valid quadruples, and Q' denotes the set of invalid triples as $Q' = \{(s, r, o', t) | o' \in E - o\}$.

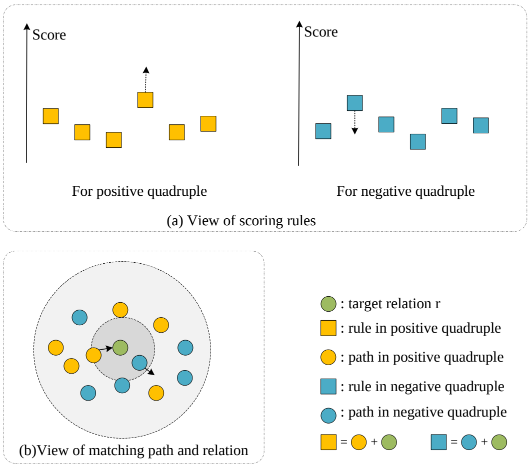

Figure 3: Training from quadruple perspective.

From the decoder, we know that score of the quadruple is determined by the rule with highest confidence it contains.

We find that all the rules drawn from the wrong quadruple must be wrong, but we cannot determine which rules drawn from the correct quadruple are correct. As shown in Figure 3(a), the training task is to make the correct quadruple score higher, that is, the soft positive rule with the highest score is regarded as a positive example to obtain a higher confidence. Similarly, for the wrong quadruple, the hard negative rule with the highest confidence is regarded as a negative example to obtain a lower confidence. From the view of matching reason path and result relation r, this task is to make the path in the positive example close to the relation r, and the path in the negative example away from the relation r, as shown in Figure 3(b). In short, this training task is to make the soft positive rules have higher confidence, make the hard negative rules that are easy to misjudgment have lower confidence, and ignore other negative rules and uncertain rules.

4.3.2 Training from rule perspective

From the fine-grained rules perspective, we add another auxiliary training task to handle rules ignored in the main task. Inspired by one-class problem, we train the model only with negative samples. By negative sampling, we can obtain a sufficient number of negative quadruplets, each of which contains multiple negative rules that can be determined to be negative. As shown in Figure 4, the relative confidence of possible positive rules is increased by decreasing confidence of negative rules, thereby improving prediction accuracy of the model.

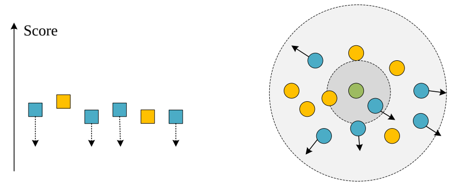

Figure 4: Training from rule perspective.

If the similarity matching function is applied to estimate confidence of the rule, the loss function here is defined as the cosine loss:

$$L_2 = \sum_{(s,r,o,t)\in Q\bigcup Q'} cosloss(p,r) \tag{11}$$

$$cosloss(p,r) = \begin{cases} 1 - cos(\mathbf{p},\mathbf{r}), & y = 1 \\ max(0, cos(\mathbf{p},\mathbf{r})), & y = -1 \end{cases} \tag{12}$$

If the concatenation combination function is applied, the loss function is the soft-margin loss:

$$L_2 = \sum_{(s,r,o,t)\in Q\bigcup Q'} log(1 + exp(l \cdot g(s,r,o,t)) \tag{13}$$

Then the overall one-class augmented matching loss is defined as:

$$L = \alpha L_1 + (1-\alpha)L_2 \tag{14}$$

where $\alpha \in [0, 1]$.

4.4 Inference

Our method can directly use the trained model to extract historical features and predict missing entity without complex rule application process. First, we select candidate entities (all entities reachable within k hops with s as the source entity node) for the query $(s, r, ?, t)$, and generate candidate quadruples based on the candidate entities. Then, we apply the trained encoder to extract relation paths for the query, form rules and assess their confidence. Finally, we score quadruples according to the confidence of rules. The candidate entity corresponding to the quadruple with the highest score is the predicted target entity.

5 Experiment

5.1 Datasets

We conduct experiments on Integrated Crisis Early Warning System[2] (ICEWS) dataset. ICEWS commonly used for temporal knowledge graph link

[2]https://dataverse.harvard.edu/dataverse/icews

Data	Entities	Relations	Training	Validation	Test	Time Granules
ICEWS14	7,128	230	63,685	13,823	13,222	365
ICEWS18	23,033	256	539,286	67,538	63,110	304
ICEWS0515	10,488	251	272,115	17,535	20,466	4,017

Table 1: Statistics of the datasets.

Method	ICEWS14				ICEWS18				ICEWS0515			
	MRR	Hits@1	Hits@3	Hits@10	MRR	Hits@1	Hits@3	Hits@10	MRR	Hits@1	Hits@3	Hits@10
DistMult	0.2767	0.1816	0.3115	0.4696	0.1017	0.0452	0.1033	0.2125	0.2873	0.1933	0.3219	0.4754
ComplEx	0.3084	0.2151	0.3448	0.4958	0.2101	0.1187	0.2347	0.3987	0.3169	0.2144	0.3574	0.5204
AnyBURL	0.2967	0.2126	0.3333	0.4673	0.2277	0.1510	0.2544	0.3891	0.3205	0.2372	0.3545	0.5046
TTransE	0.1343	0.0311	0.1732	0.3455	0.0831	0.0192	0.0856	0.2189	0.1571	0.0500	0.1972	0.3802
TA-DistMult	0.2647	0.1709	0.3022	0.4541	0.1675	0.0861	0.1841	0.3359	0.2431	0.1458	0.2792	0.4421
DE-SimplE	0.3267	0.2443	0.3569	0.4911	0.1930	0.1153	0.2186	0.3480	0.3502	0.2591	0.3899	0.5275
TNTComplEx	0.3212	0.2335	0.3603	0.4913	0.2123	0.1328	0.2402	0.3691	0.2754	0.1952	0.3080	0.4286
CyGNet	0.3273	0.2369	0.3631	0.5067	0.2493	0.1590	0.2828	0.4261	0.3497	0.2567	0.3909	0.5294
RE-NET	0.3828	0.2868	0.4134	0.5452	0.2881	0.1905	0.3244	0.4751	0.4297	0.3126	0.4685	0.6347
xERTE	0.4079	0.3270	0.4567	0.5730	0.2931	0.2103	0.3351	0.4648	0.4662	0.3784	0.5231	0.6392
TLogic	0.4304	0.3356	0.4827	0.6123	0.2982	0.2054	0.3395	0.4853	0.4697	0.3621	0.5313	0.6743
RE-GCN	0.4435	0.3351	0.5081	0.6316	0.3484	0.2309	0.3983	0.5816	0.4923	0.3824	0.5571	0.7054
ALRE-IR	**0. 5401**	**0.4279**	**0.6116**	**0.7179**	**0.3841**	**0.2566**	**0.4372**	**0.6100**	**0.6018**	**0.4897**	**0.6777**	**0.7750**
ALRE-IR w/i CE	0.6384	0.5380	0.7091	0.7907	0.4537	0.3778	0.4810	0.6661	0.6479	0.5588	0.7048	0.7803

Table 2: Performance comparison for entity prediction.

prediction, which contains international event information. We select three subsets in ICEWS dataset, namely ICEWS0515, which contains data from 2005 to 2015, ICEWS14 which contains data in 2014, and ICEWS18 which contains data in 2018. We divide each dataset into training set, validation set and test set, and for fair comparison, we used the data splits provided by Liu et al. (2022). Table 1 provides statistics of all datasets used.

5.2 Baselines

To demonstrate effectiveness of our proposed ALRE-IR model, we compare experimental results with a wide selection of static models and temporal models.

Static models. We select some static knowledge graph representation learning models that ignore time information, including DistMult (Yang et al., 2015), ComplEX (Trouillon et al., 2016), and AnyBURL (Meilicke et al., 2020).

Temporal models. We also compare some temporal reasoning models of knowledge graphs, including TTransE (Leblay and Chekol, 2018), DE-SimplE (Goel et al., 2020), TNTComplEx (Lacroix et al., 2020), TA-DistMult (Garcia-Duran et al., 2018),RE-NET (Jin et al., 2020), CyGNet (Zhu et al., 2021), xERTE (Han et al., 2021), TLogic (Liu et al., 2022) and RE-GCN (Li et al., 2021). For RE-GCN (Li et al., 2021), we reproduce the experiments. And for other baselines, we list the results reported in TLogic (Liu et al., 2022).

Due to time urgency of news events, related events may appear on the same day. Datasets that roughly label temporal information in units of days hinder the model's ability to extract historical information. Therefore, we also try to use partial events in current timestamp to provide some hints during inference, named as ALRE-IR w/i CE. We sort events within the same timestamp according to the order in which they appear in the dataset, mask the later events, and use the earlier events as known historical events to assist in reasoning.

5.3 Results

Experimental results are shown in Table 2. All models perform best on ICEWS0515, followed by ICEWS14, and the worst on ICEWS18. It can be seen from Table 1 that the number of entities and events in ICEWS18 dataset is large, so the TKG structure formed is complex and dense, which brings a lot of noise to inference. In contrast, the TKG composed of ICEWS0515 is easier to handle.

For comparison models, three static inference models DistMult, ComplEX, and AnyBURL do not consider temporal information and thus perform worst. TTransE, DE-SimplE, TNTComplEx and TA-DistMult are interpolation inference models, which cannot handle events in future timestamps and perform poorly. RE-NET and CyGNet fail to make predictions on entities which do not exist in the training set. xERTE achieves better performance than RE-NET and CyGNet, since it extracts historical subgraph according to the query and performs attention propagation to reason on the sub-

Model	ALRE-IR		RE-GCN	ALRE-IR		RE-GCN
Test	ICEWS14			ICEWS0515		
Train	ICEWS14	ICEWS0515	ICEWS14	ICEWS0515	ICEWS14	ICEWS0515
MRR	0.5401	0.5056	0.4435	0.6018	0.5867	0.4923
Hits@1	0.4279	0.3894	0.3351	0.4897	0.4650	0.3824
Hits@3	0.6116	0.5757	0.5081	0.6777	0.6659	0.5571
Hits@10	0.7179	0.6917	0.6316	0.7750	0.7920	0.7054

Table 3: Zero-shot reasoning where rules learned on train dataset are transferred and applied to test dataset.

graph. The two best performing models are logical rule-based TLogic and embedding-based RE-GCN. Our proposed ALRE-IR outperforms the above two models on all datasets. The measure used in TLogic to assess the confidence of logical rules are designed based on statistical methods. Instead, we use the learned path embeddings to evaluate the rule confidence according to causal logic. The distance between embedding vectors can well reflect the similarity between paths and relations. RE-GCN leverages graph convolutional networks to learn evolutional representations of entities and relations, achieving better performance than TLogic. Both TLogic and our proposed ALRE-IR can transfer the trained model for inductive reasoning on datasets with common relation vocabulary, but RE-GCN cannot do this.

Logical rule-based methods can effectively predict events that contain rules mined from the training set, but are slightly less effective for unseen rules. As shown in Table 2, hits@1 of RE-GCN on ICEWS14 is slightly lower than TLogic, but Hits@3 and Hits@10 are higher than TLogic. Embedding-based RE-GCN learns entity evolutional representations and predicts events based on distances between vector representations. The learned vector representations can well reflect the latent relationships between entities, so that correct quadruples can get higher scores. But they ignore the important logical relationships contained in the knowledge graph, making it difficult to obtain accurate prediction. Our proposed model combines advantages of the two methods, learns causal path embedding to mine underlying logic, improves the model's accurate prediction ability, and enhances robustness to unseen rules, thereby achieving better performance.

Furthermore, the outstanding performance of ALRE-IR w/i CE shows that there is a strong correlation between events occurring within the same timestamp, which can assist the model in making real-time predictions.

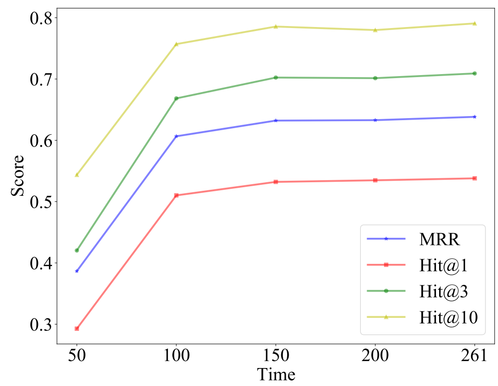

Figure 5: Result on ICEWS14 dataset under different scales of training samples.

5.4 Zero-shot reasoning

Our proposed ALRE-IR model can be transferred to any new dataset that shares a common relationship with the training dataset for zero-shot reasoning. To evaluate the zero-shot reasoning performance of ALRE-IR, we conduct experiments on ICEWS0515 and ICEWS14, and the results are shown in Table 3. The ALRe-IR model trained on ICEW0515 is applied to ICEWS14 for reasoning. The prediction performance is slightly worse than the ALRE-IR model trained on ICEWS14, but still better than the best baseline RE-GCN. Similar performance is achieved when the model is trained on ICEWS14 and tested on ICEW0515.

5.5 Proportions of the training data

We evaluate performance of the proposed ALRE-IR model under different scales of training samples, and the results are shown in Figure 5. We divide the training set by timestamp and evaluate model's performance when trained with events within the previous 50, 100, 150, 200, and 261 timestamps (full training set), respectively. When trained with the events only within the previous 50 timestamps, the model underfitted and performed poorly. But when we train the model with the previous 100 timestamp events, it achieves similar performance to the model trained with the full training set. It shows that our model can achieve good performance with only a small number of training samples.

5.6 Error analysis

To analyze errors of our proposed model in TKG reasoning, we randomly sample 100 inaccurately predicted test quadruples and summarize three types of errors. (1) Same relation paths: The model scores the quadruples based on similarity between the historical relationship path and the current relationship. When the same reasonable historical relationship path is mined for different quadruples, they will obtain the same score. (2) Time insensitivity: Due to insufficient use of temporal information, time of the event occurrence is incorrectly predicted, that is, future events are predicted at the current moment. (3) Immediate response events: Urgent related events may occur on the same day. Time interval of events in the dataset is one day, which prevents us from capturing key historical events that occurred on the same day.

Due to space limitations, we report more results including implementation, detailed analysis and case study in Appendices.

6 Conclusion

We propose an interpretable model for temporal knowledge graph reasoning called ALRE-IR. It can autonomously extract and assess rules based on historical features, and make prediction with rules. We design training tasks from a coarse-grained quadruple perspective and a fine-grained rule perspective, respectively, and propose a one-class augmented matching loss for optimization. ALRE-IR can be transferred to perform zero-shot reasoning on any new dataset with common relation vocabulary. Experimental results demonstrate that our proposed ALRE-IR performs better than the state-of-the-art baselines.

Limitations

Although our proposed ALRE-IR model has shown better performance than state-of-the-art baselines, there are some limitations. We do not fully consider temporal information when mining logical rules. We only focus on logical causality of historical paths and current events, while ignoring specific time when the "result" event occurred. Regarding this issue, we consider adding temporal information in path encoding, taking time difference between the relation edge in the path and current relation edge as temporal feature, and encoding both the semantic feature and temporal feature to improve prediction performance.

Acknowlegement

This work was supported in part by the National Natural Science Foundation of China under Grants 61872296, 61772429, and U20B2065, and in part by the MOE (Ministry of Education in China) Project of Humanities and Social Sciences under Grant18YJC870001.

References

Trapit Bansal, Da-Cheng Juan, Sujith Ravi, and Andrew McCallum. 2019. A2n: Attending to neighbors for knowledge graph inference. In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pages 4387–4392.

Johan Bollen, Huina Mao, and Xiaojun Zeng. 2011. Twitter mood predicts the stock market. *Journal of computational science*, 2(1):1–8.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. *Advances in neural information processing systems*, 26.

Elizabeth Boschee, Jennifer Lautenschlager, Sean O'Brien, Steve Shellman, James Starz, and Michael Ward. 2015. Icews coded event data. *Harvard Dataverse*, 12.

Shib Sankar Dasgupta, Swayambhu Nath Ray, and Partha Talukdar. 2018. Hyte: Hyperplane-based temporally aware knowledge graph embedding. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pages 2001–2011.

Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. 2018. Convolutional 2d knowledge graph embeddings. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.

Xin Dong, Evgeniy Gabrilovich, Geremy Heitz, Wilko Horn, Ni Lao, Kevin Murphy, Thomas Strohmann, Shaohua Sun, and Wei Zhang. 2014. Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 601–610.

Alberto Garcia-Duran, Sebastijan Dumančić, and Mathias Niepert. 2018. Learning sequence encoders for temporal knowledge graph completion. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4816–4821.

Rishab Goel, Seyed Mehran Kazemi, Marcus Brubaker, and Pascal Poupart. 2020. Diachronic embedding for temporal knowledge graph completion. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 3988–3995.

Simon Gottschalk and Elena Demidova. 2018. Eventkg: a multilingual event-centric temporal knowledge graph. In *European Semantic Web Conference*, pages 272–287. Springer.

Simon Gottschalk and Elena Demidova. 2019. Eventkg–the hub of event knowledge on the web–and biographical timeline generation. *Semantic Web*, 10(6):1039–1070.

Zhen Han, Peng Chen, Yunpu Ma, and Volker Tresp. 2021. Explainable subgraph reasoning for forecasting on temporal knowledge graphs. In *9th International Conference on Learning Representations, ICLR*.

Aidan Hogan, Eva Blomqvist, Michael Cochez, Claudia d'Amato, Gerard de Melo, Claudio Gutierrez, Sabrina Kirrane, José Emilio Labra Gayo, Roberto Navigli, Sebastian Neumaier, et al. 2021. Knowledge graphs. *Synthesis Lectures on Data, Semantics, and Knowledge*, 12(2):1–257.

Wen Hua, Kai Zheng, and Xiaofang Zhou. 2015. Microblog entity linking with social temporal context. In *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, pages 1761–1775.

Woojeong Jin, Meng Qu, Xisen Jin, and Xiang Ren. 2020. Recurrent event network: Autoregressive structure inference over temporal knowledge graphs. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6669–6683.

Gizem Korkmaz, Jose Cadena, Chris J Kuhlman, Achla Marathe, Anil Vullikanti, and Naren Ramakrishnan. 2015. Combining heterogeneous data sources for civil unrest forecasting. In *Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015*, pages 258–265.

Timothee Lacroix, Guillaume Obozinski, and Nicolas Usunier. 2020. Tensor decompositions for temporal knowledge base completion. In *8th International Conference on Learning Representations*.

Julien Leblay and Melisachew Wudage Chekol. 2018. Deriving validity time in knowledge graph. In *Companion Proceedings of the The Web Conference 2018*, pages 1771–1776.

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.

Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. 2015. Learning entity and relation embeddings for knowledge graph completion. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, pages 2181–2187.

Yushan Liu, Yunpu Ma, Marcel Hildebrandt, Mitchell Joblin, and Volker Tresp. 2022. Tlogic: Temporal logical rules for explainable link forecasting on temporal knowledge graphs. *Proceedings of the AAAI Conference on Artificial Intelligence*.

Kangqi Luo, Fengli Lin, Xusheng Luo, and Kenny Zhu. 2018. Knowledge base question answering via encoding of complex query graphs. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2185–2194.

Christian Meilicke, Melisachew Wudage Chekol, Manuel Fink, and Heiner Stuckenschmidt. 2020. Reinforced anytime bottom up rule learning for knowledge graph completion. *arXiv preprint arXiv:2004.04412*.

Pablo N Mendes, Max Jakob, Andrés García-Silva, and Christian Bizer. 2011. Dbpedia spotlight: shedding light on the web of documents. In *Proceedings of the 7th international conference on semantic systems*, pages 1–8.

Bonan Min, Ralph Grishman, Li Wan, Chang Wang, and David Gondek. 2013. Distant supervision for relation extraction with an incomplete knowledge base. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 777–782.

Sathappan Muthiah, Bert Huang, Jaime Arredondo, David Mares, Lise Getoor, Graham Katz, and Naren Ramakrishnan. 2015. Planned protest modeling in news and social media. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*. Citeseer.

Dai Quoc Nguyen, Tu Dinh Nguyen, Dat Quoc Nguyen, and Dinh Phung. 2018. A novel embedding model for knowledge base completion based on convolutional neural network. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 327–333.

Maximilian Nickel, Kevin Murphy, Volker Tresp, and Evgeniy Gabrilovich. 2015. A review of relational machine learning for knowledge graphs. *Proceedings of the IEEE*, 104(1):11–33.

Pouya Ghiasnezhad Omran, Kewen Wang, and Zhe Wang. 2019. Learning temporal rules from knowledge graph streams. In *AAAI Spring Symposium: Combining Machine Learning with Knowledge Engineering*.

Lawrence Phillips, Chase Dowling, Kyle Shaffer, Nathan Hodas, and Svitlana Volkova. 2017. Using social media to predict the future: A systematic literature. *arXiv preprint arXiv:1706.06134*.

Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne van den Berg, Ivan Titov, and Max Welling. 2018. Modeling relational data with graph convolutional networks. In *European semantic web conference*, pages 593–607. Springer.

Baoxu Shi and Tim Weninger. 2018. Open-world knowledge graph completion. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.

Alessio Signorini, Alberto Maria Segre, and Philip M Polgreen. 2011. The use of twitter to track levels of disease activity and public concern in the us during the influenza a h1n1 pandemic. *PloS one*, 6(5):e19467.

Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. 2019. Rotate: Knowledge graph embedding by relational rotation in complex space. In *7th International Conference on Learning Representations*.

Ye Tao, Ying Li, and Zhonghai Wu. 2021. Temporal link prediction via reinforcement learning. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 3470–3474. IEEE.

Kristina Toutanova and Danqi Chen. 2015. Observed versus latent features for knowledge base and text inference. In *Proceedings of the 3rd workshop on continuous vector space models and their compositionality*, pages 57–66.

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.

Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. 2016. Complex embeddings for simple link prediction. In *International conference on machine learning*, pages 2071–2080. PMLR.

Quan Wang, Zhendong Mao, Bin Wang, and Li Guo. 2017. Knowledge graph embedding: A survey of approaches and applications. *IEEE Transactions on Knowledge and Data Engineering*, 29(12):2724–2743.

Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. 2014. Knowledge graph embedding by translating on hyperplanes. In *Proceedings of the AAAI conference on artificial intelligence*, volume 28.

Jiapeng Wu, Yishi Xu, Yingxue Zhang, Chen Ma, Mark Coates, and Jackie Chi Kit Cheung. 2021. Tie: A framework for embedding-based incremental temporal knowledge graph completion. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '21, page 428–437.

Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. 2015. Embedding entities and relations for learning and inference in knowledge bases. In *3rd International Conference on Learning Representations*.

Scott Wen-tau Yih, Ming-Wei Chang, Xiaodong He, and Jianfeng Gao. 2015. Semantic parsing via staged query graph generation: Question answering with knowledge base. In *Proceedings of the Joint Conference of the 53rd Annual Meeting of the ACL and the 7th International Joint Conference on Natural Language Processing of the AFNLP*.

Daojian Zeng, Kang Liu, Yubo Chen, and Jun Zhao. 2015. Distant supervision for relation extraction via piecewise convolutional neural networks. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 1753–1762.

Zhao Zhang, Fuzhen Zhuang, Hengshu Zhu, Zhiping Shi, Hui Xiong, and Qing He. 2020. Relational graph neural network with hierarchical attention for knowledge graph completion. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 9612–9619.

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.

A Appendix

A.1 Implementations

We randomly initialize relation embeddings with dimension of 200. The initial hyperparameter of α is set to 0.5 and increases by 0.1 every training epoch until it reaches 1. The maximum length k of the rules is set as 3, and the optimal historical event intervals m on the ICEWS0515, ICEWS14 and ICEWS18 datasets are set to 5, 3, and 3, respectively. We use Adam optimizer to optimize all parameters, and the initial learning rate is set as 0.001. We use early stopping to avoid overfitting. We train the model for 200 epochs and stop training if the validation loss does not decrease for 10 consecutive epochs.

We adopt a time-aware filtering strategy (Han et al., 2021) to filter out the quadruples valid at current timestamp among the candidate negative quadruples. When extracting rules, we treat the knowledge graph as an undirected graph. MRR, Hit@1, Hit@3 and Hit@10 are employed as the metrics.

Method	ICEWS14				ICEWS0515			
	MRR	Hits@1	Hits@3	Hits@10	MRR	Hits@1	Hits@3	Hits@10
ALRE-IR w/i SM	0.5401	0.4279	0.6116	0.7179	0.6018	0.4897	0.6777	0.7750
ALRE-IR w/i CC	0.3886	0.2927	0.4264	0.5477	0.4332	0.3292	0.4824	0.5920

Table 4: Results with different confidence estimation functions.

Method	ICEWS14				ICEWS0515			
	MRR	Hits@1	Hits@3	Hits@10	MRR	Hits@1	Hits@3	Hits@10
ALRE-IR w/i M	0. 5401	0.4279	0.6116	0.7179	0.6018	0.4897	0.6777	0.7750
ALRE-IR w/i A	0.4196	0.3405	0.4608	0.5388	0.4610	0.3739	0.5036	0.5969

Table 5: Results with different score functions.

A.2 Detailed Analysis

The results in this section are obtained on datasets ICEWS14 and ICEWS0515, with similar results on the ICEWS18 dataset.

A.2.1 Confidence estimation function

In this paper, two rule confidence evaluation methods, similarity matching and concatenation combination, are proposed. In order to evaluate these two methods, we conduct experiments on ICEWS14 and ICEWS0515 datasets respectively, and the results are reported in Table 4. ALRE-IR w/i SM and ALRE-IR w/i CC represent models that employ similarity matching and concatenation combination to evaluate rule confidence, respectively. It can be seen from Table 4 that ALRE-IR w/i CC performs significantly worse than ALRE-IR w/i SM on both datasets. It indicates that similarity-based measures can better reflect causal association between paths and relations, so we adopt similarity matching on three datasets to evaluate the confidence of rules.

A.2.2 Score function

When scoring a quadruple, the score function introduced in this paper takes highest confidence of all rules as the score of the quadruple, named ALRE-IR w/i M. We also try another way, that is, averaging all path embeddings to get a global path embedding, and taking confidence of the rule composed of global paths and relations as the score of the quadruple, named ALRE-IR w/i A. We conduct experiments on ICEWS14 and ICEWS0515 datasets, and the results are shown in Table 5. It is clearly to see that averaging all path embeddings works poorly, because not all paths contribute to the current event.

A.3 Case study

Table 6 shows prediction results for two queries (Uhuru Muigai Kenyatta, Demand,?,t) and (South Korea, Sign formal agreement,?,t) on the ICEWS14 dataset. The table shows the paths from the subject entity to candidate object entities, the rules composed of relation paths, and the scores of the corresponding rules. ALRE-IR aims to find the target entity corresponding to the most reasonable rule.

Query	Path and rule	Score	Target entity
(Uhuru Muigai Kenyatta, Demand,?,t)	P: Citizen (Kenya) $\xrightarrow[t-1]{Demand\,meeting}$ Uhuru Muigai Kenyatta R: Demand meeting⇒ Demand^{-1}	0.2699	Citizen (Kenya) (√)
	P: Uhuru Muigai Kenyatta $\xrightarrow[t-1]{Threaten}$ Citizen(Kenya) R: Threaten⇒ Demand	0.0597	
	P: Police(Kenya) $\xrightarrow[t-1]{Demand}$ Citizen(Kenya) $\xrightarrow[t-1]{Demand\,meeting}$ Uhuru Muigai Kenyatta R: Demand|Demand meeting⇒ Demand^{-1}	0.0356	Police(Kenya)
	P: William Ruto $\xrightarrow[t-2]{makeanappeal}$ Citizen (Kenya) $\xrightarrow[t-1]{Demand\,meeting}$ Uhuru Muigai Kenyatta R: Make an appeal |Demand meeting⇒ Demand^{-1}	-0.1009	William Ruto
(South Korea, Sign formal agreement,?,t)	P: South Korea $\xrightarrow[t-1]{Express\,intent\,to\,cooperate}$ China R: Express intent to cooperate⇒Sign formal agreement	0.4960	China(√)
	P: South Korea $\xrightarrow[t-2]{Express\,intent\,to\,cooperate\,economically}$ China R: Express intent to cooperate economically⇒ Sign formal agreement	0.2850	
	P: South Korea $\xrightarrow[t-2]{Engage\,in\,negotiation}$ China R: Engage in negotiation⇒Sign formal agreement	0.1165	
	P: South Korea $\xrightarrow[t-2]{Express\,intent\,to\,provide\,economic\,aid}$ International Government Organizations R: Express intent to provide economic aid⇒ Sign formal agreement	0.3470	International Government Organizations
	P: South Korea $\xrightarrow[t-2]{Express\,intent\,to\,provide\,humanitarian\,aid}$ Sierra Leone R: Express intent to provide humanitarian aid⇒ Sign formal agreement	0.2140	Sierra Leone
	P: North Korea $\xrightarrow[t-1]{Occupy\,territory}$ South Korea R: Occupy territory⇒Sign formal agreement^{-1}	0.0065	North Korea

Table 6: Entity prediction visualization on ICEWS14. Since the dataset does not provide accurate time, we use t to denote the time of the event.