Retrieval-Augmented Zero-Shot Generalization of Logic Rules to Unseen Relations for Temporal Knowledge Graph Forecasting

Anonymous ACL submission

Abstract

Link prediction leverages a temporal knowledge graphs (TKGs) to forecast relationships between entities over time. Although most existing research focuses on the transductive setting, assuming that all entities and relations are known in advance, due to the dynamic nature of TKGs, new, previously unseen relations frequently emerge, requiring models with strong generalization capabilities. We propose an inductive learning model, Retrieval-Enhanced Extrapolated Logic (REEL), designed to augment a rule-based link forecasting model on unseen relations. Given a query with an unseen relation, REEL discovers highly correlated logic rules within the TKG and uses these rules for link forecasting. REEL first finds the most relevant existing relations with document similarity. Then, the associated logic rules are identified and weighted based on their similarity. Finally, REEL applies the weighted logic rules to the unseen relation. Experimental results demonstrate that REEL consistently and significantly outperforms existing methods for TKGF. Notably, on the zero-shot variation of the DuEE dataset, REEL's MRR is more than double the second-best MRR score.

1 Introduction

Temporal knowledge graph forecasting (TKGF), predicting future events based on historical data, is a critical task in many real-world domains such as healthcare (Postiglione et al., 2024), social media (Jia et al., 2024; Gao et al., 2022), and finance (Ibrahim et al., 2022).

Existing research on TKGF (Xu et al., 2023; Zhu et al., 2021; Li et al., 2021; Omran et al., 2021; Qu et al., 2021; Cheng et al., 2023; Xu et al., 2022; Xiong et al.) primarily focuses on the transductive setting, which assumes that both entities and relations have been seen in training. Embedding-based approaches (Xu et al., 2023; Zhu et al., 2021; Li et al., 2021) reason in the embedding space, which, while effective, are often opaque and lack interpretability. Rule-based approaches (Omran et al., 2021; Qu et al., 2021; Cheng et al., 2023; Xu et al., 2022; Xiong et al.) utilize logic rules to rank candidate entities, offering greater interpretability.

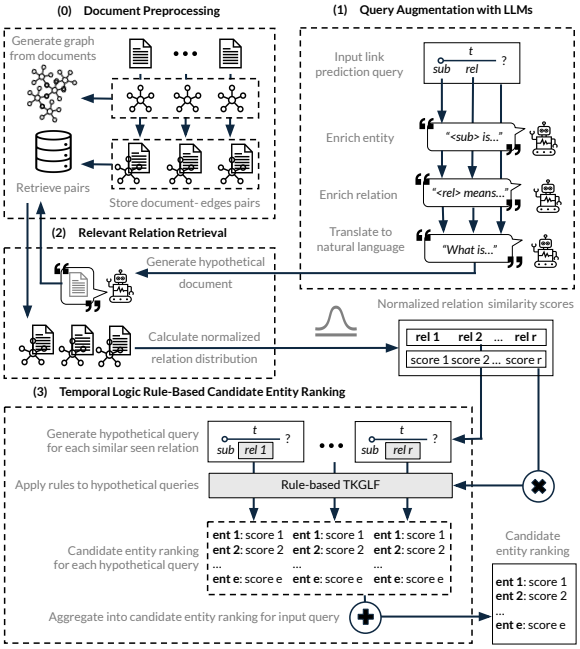

Figure 1: Overview of REEL

Real world TKGs are constantly evolving and new relations frequently emerge. However, link forecasting for unseen relations poses a significant challenge to existing link forecasting methods which struggle to generalize. Notably, rule-based methods are limited by their reliance on rules learned from training. For queries with unseen relations, rule-based methods are forced to resort to naive baselines.

To tackle this problem, in this paper, we propose to apply the rules learned for existing relations to unseen relations for enhancing its generalization ability. Generally speaking, a TKG is created by identifying events described in a text and writ-

ing the event as a TKG edge (Knez and Žitnik, 2023). Our method Retrieval-Enhanced Extrapolated Logic (REEL) leverages this set of documents and graph edges extracted from the documents, referred to as the document-edges pair set, which serves as an additional source of information. We first use an LLM to generate a hypothetical document for the query, and then based on this hypothetical document, we are able to find the most similar documents from the document-edges pair set. The normalized frequency of the relations in the associated graph edges are considered as the approximate similarity score between each candidate relation and the query relation, and the set of relations from the associated graph edges are considered as the relations that are most similar to the relation in the query. Once these relations are identified, hypothetical queries are generated by replacing the unseen relation with the relations from the graph edges and are used as the input query for a traditional rule-based approach, allowing candidate ranking for each of the hypothetical queries. Finally, these rankings are aggregated into the overall candidate ranking taking into consideration of the relation similarity scores. Figure 1 presents an overview of the main components.

To our best knowledge, this research represents the first work to (1) enable rule-based link forecasting for unseen relations and (2) utilize a parallel corpus of natural language documents for the purposes of zero-shot generalization in the link forecasting task.

The main contributions of this paper[1] are:

1. **Approach.** We introduce REEL, an framework to perform interpretable TKG forecasting on unseen relations by applying rules learned for existing relations to unseen relations by utilizing a hypothetical document generated by an LLM.

2. **Datasets.** We curate three new versions of datasets (zs-DuEE, zs-Wiki, and zs-GDELT) to evaluate zero-shot TKG forecasting on both unseen relations and unseen entities with a parallel corpus of the source documents.

3. **Evaluation.** We perform comprehensive experiments to evaluate the performance of embedding-based and rule-based approaches for zero-shot TKG forecasting and show REEL consistently has the highest MRR on the unseen relations, improving MRR on unseen relations by up to 1,600% over the unaugmented rule-based approach.

[1]Code can be found at https://anonymous.4open.science/r/REEL-E141.

2 Related Work

2.1 Knowledge Graph Completion

Link forecasting over TKGs mainly fall into two classes of approaches: embedding-based and logic-rule-based. Embedding-based approaches are more common (Xu et al., 2023; Zhu et al., 2021; Li et al., 2021). These approaches learn embeddings for the entities and relations in the knowledge graph and use them to answer queries over the knowledge graph. The other class of approaches are those based on logic rules (Omran et al., 2021; Qu et al., 2021; Cheng et al., 2023; Xu et al., 2022; Xiong et al.). These methods extract logic rules which are used to answer knowledge graph queries.

Logic rules are formal expressions that capture relationships between entities mined from walks in the training TKG. In the context of TKGs, the rules capture common sequences of relations that occur in the TKG with the entities being free variables. Due to the structure of these rules, rules can be directly applied to make inferences about *unseen entities* without requiring re-training (Liu et al., 2022).

Although rule-based TKG methods can apply their learned logic to *unseen entities* within familiar relational structures, they lack the mechanism to reason about *unseen relations* that are not part of their rule set because these rules are constructed to predict the tail entity of a link forecasting query based on walks from the training TKG (Zeng et al., 2023). Furthermore, unlike embedding-based methods which can leverage semantic information, rule-based approaches have no inherent mechanism to understand or generalize to novel relation types. As a result, even if the TKG is updated with the new relations and entities, a rule-based TKGF method would be unable to generalize to unseen relations even thought it will be able to generalize to unseen entities.

The majority of literature in inductive link forecasting focus on generalizing to unseen entities (Ding et al., 2023b; Wang et al., 2021; Shah et al., 2019). These approaches typically leverage information such as entity name, textual descriptions, or attributes using word embeddings (Shah et al., 2019) or a language model (Wang et al., 2021)

to perform generalization. Among the works in generalization to unseen relations, they are almost exclusively devoted to improving generalizability of embedding-based methods. Furthermore, the majority of the works require at least one example of the new relation which is different from the zero-shot setting we investigate in our approach (Ding et al., 2023a; Mirtaheri et al., 2021). A few works leverage semantic features via an LLM or GAN to perform zero-shot relational TKG forecasting (Ding et al., 2024; Qin et al., 2020). However, this remains an under-explored space. Furthermore, these works apply only to embedding-based approaches, which lacks interpretability.

2.2 Reasoning over Knowledge Graphs with Semantic Information

Large language models (LLMs) have shown strong performance on a wide variety of tasks (Wei et al., 2022). Furthermore, since they have been trained on diverse text data, LLMs also act as strong general knowledge bases (Petroni et al., 2019). There have been several works leveraging LLMs in conjunction with TKGs (Han et al., 2023; Ding et al., 2024). Han et al. leverages LLMs to generate text that is used to enhance TKG embeddings, but does not focus on the problem setting of link forecasting and assumes all entities and relations are seen during training. Ding et al. focuses on the same problem setting as us, but the key difference is that it only applies to embedding-based methods, which lacks interpretability, and their method does not introduce external data to improve performance.

3 Problem Definition

In this section, we formally define the problem of temporal knowledge graph forecasting and introduce key concepts related to it.

Let $\mathcal{E}$, $\mathcal{R}$, and $\mathcal{T}$ represent the set of entities, relations, and timestamps respectively.

Temporal Knowledge Graph (TKG). A temporal knowledge graph $\mathcal{G} \subset \mathcal{E} \times \mathcal{R} \times \mathcal{E} \times \mathcal{T}$ consists of a set of facts in the form of quadruples (s, r, o, t), where $s, o \in \mathcal{E}$, $r \in \mathcal{R}$, and $t \in \mathcal{T}$. Each quadruple represents a directed edge in the graph: s is the subject entity, r is the relation connecting the subject s to the object entity o, and t is the timestamp at which this relationship holds. The timestamp t can be measured in units such as hour, day, and year (Liao et al., 2024; Liu et al., 2022).

Given a TKG, the task of link forecasting aims to predict the missing entity in a future fact, typically in the form $(s, r, ?, t)$ or $(?, r, o, t)$, where t is a future timestamp. The prediction is based on historical data observed in the TKG, leveraging both the structural relationships between entities and their temporal evolution over time. In this paper, we mainly focus on predicting missing objects in queries of the form $(s, r, ?, t)$, as queries of the form $(?, r, o, t)$ can be equivalently expressed as $(o, r^{-1}, ?, t)$, where r^{-1} denotes the inverse of relation r.

To effectively predict missing entities in temporal link forecasting, it is important to utilize the structural information inherent in the TKG. Temporal logic rules can provide valuable insights by capturing the relationships and temporal patterns that govern entity interactions over time, which are defined as follow.

Temporal Logic Rules. A cyclic temporal logic rule l of length $j \in \mathbb{N}$ is defined as $(e_1, r_h, e_{j+1}, t_{j+1}) \leftarrow \wedge_{i=1}^{j}(e_i, r_i, e_{i+1}, t_i)$ where $e_1, ..., e_{j+1} \in \mathcal{E}$, $r_h, r_1, ...r_j \in \mathcal{R}$ $t_1, ..., t_{j+1} \in \mathcal{T}$, and the following temporal constraint holds: $t_1 \leq t_2 \leq \cdots \leq t_j \leq t_{j+1}$ (Liao et al., 2024; Liu et al., 2022).

The left-hand side of the rule is referred to as the rule head, while the right-hand side is known as the rule body.

A temporal logic rule indicates that if the rule body and all associated entity, relation, and temporal constraints are satisfied, then the rule head is valid for the future timestamp t_{j+1}. For example, the temporal logic rule (e_1, *give present*, e_2, t_3) $\leftarrow$ (e_1, *give present*, e_2, t_1) $\wedge$ (e_2, *give present*, e_1, t_2) states if entity 1 has previously given a present by entity 2, and subsequently had been given a present by entity 2, the entity 1 will give a present to entity 2 in the future.

A rule grounding is a specific instance of a rule, where the variables are replaced with concrete values so that it represents a valid path over the TKG that follows the temporal constraints of the rule. For the temporal logic rule above, a potential grounding could be (*Alice*, *give present*, *Bob*, t_2) $\leftarrow$ (*Alice*, *give present*, *Bob*, *2022-01-01*) $\wedge$ (*Bob*, *give present*, *Alice*, *2023-01-01*) where $e_1 = Alice$ and $e_2 = Bob$. t_2 is not replaced since we only know $t_2 \geq$ 2023-01-01. The sequence of facts that make up the body of the rule grounding e.g. (*Alice*, *give present*, *Bob*, *2022-01-01*) $\wedge$ (*Bob*, *give present*, *Alice*, *2023-01-01*) is called the rule body grounding.

In practice, logical rules are often violated due to the inherent incompleteness and noise in temporal knowledge graphs. A more practical solution is to assign each rule l a confidence score, conf(l), which estimates its likelihood of being correct. This probabilistic approach allows for greater flexibility and a more accurate interpretation of logical systems, accommodating the uncertainties in real-world data. In this paper, for a given relation r_h, we define $\mathcal{L}_{r_h}$ as the set of all possible logic rules that has r_h as the relation in the rule head.

Traditional TKGF methods often assume that during the testing phase, all relations are known from the training data. However, this assumption does not hold in real-world scenarios. Due to the dynamic nature of TKGs, new relations frequently emerge over time, making it difficult for traditional methods to generalize or adapt to these unseen relations. This limitation presents significant challenges for practical applications. To address this issue, in this paper, we focus on the task of TKG link forecasting for unseen relations during the test phase. Our formal problem definition is presented below.

Task Setting.

Assume a ground-truth TKG $\mathcal{G}_{gt} \subset \mathcal{E} \times \mathcal{R} \times \mathcal{E} \times \mathcal{T}$ where $\mathcal{E}$, $\mathcal{R}$, and $\mathcal{T}$ are split into two disjoint sets of seen ($*_{in}$) and unseen ($*_{out}$) elements ($\mathcal{R} = \mathcal{R}_{in} \cup \mathcal{R}_{out}$; $\mathcal{R}_{in} \cap \mathcal{R}_{out} = \emptyset$) and all timestamps in $\mathcal{T}_{in}$ are earlier than $\mathcal{T}_{out}$ ($\forall t_1 \in \mathcal{T}_{in}, \forall t_2 \in \mathcal{T}_{out}, t_1 < t_2$).

The problem setting we focus on is TKG link forecasting under the zero-shot setting where the link forecasting query $(s_q, r_q, ?, t_q)$ contains an entity that may be unseen in training, $s_q \in \mathcal{E}$, a relation that is not seen in training, $r_q \in \mathcal{R}_{out}$, and occurs on a timestamp unseen in training, $t_q \in \mathcal{T}_{out}$. The models predict the missing entity o_q using a graph and a parallel document-edges pair set.

The graph that the models have access to contains events that had occurred prior to this query $\mathcal{G} := \{(s, r, o, t) \in \mathcal{G}_{gt} | s, o \in \mathcal{E}_{in}, r \in \mathcal{R}_{in}, t \in \mathcal{T}_{in} < t_q\}$.

The parallel document-edges pair set P is the collection of documents $\mathcal{D}$ that was used to generate the graph and the associated edges $P := \{(d, \mathcal{G}_d) | d \in \mathcal{D}, \mathcal{G}_d \subseteq \mathcal{G}\}$ where $d \in \mathcal{D}$ is the natural language text document and $\mathcal{G}_d \subseteq \mathcal{G}$ is the set of facts from $\mathcal{G}$ that were created from this document d. If $d =$ *"On January 1st, 2022, Alice gave Bob a present. The next year, Bob gave Alice a present,"* then one set of edges that may be extracted from this document is $\mathcal{G}_d = \{$(*Alice*, *give present*, *Bob*, *2022-01-01*), (*Bob*, *give present*, *Alice*, *2023-01-01*)$\}$.

4 Proposed Approach

We introduce REEL, a rule-based TKGF augmentation framework utilizes large language models (LLMs) to boost zero-shot performance by transferring knowledge from seen relations. REEL works within the RAG framework (Lewis et al., 2020) to find relevant seen relations. It selects document-edge pairs that are similar to a hypothetical document generated by an LLM based on the link forecasting query. These pairs help identify and score relations based on their similarity to the unseen relation. REEL then uses the learned rules from the seen relations to rank candidates for the unseen relation. This framework consists of a preprocessing stage (Section 4.1), a query augmentation stage (Section 4.2), a retrieval stage (Section 4.3) and a ranking stage (Section 4.4).

4.1 Preprocessing

TKGs are typically constructed based on a set of documents $\mathcal{D}$. Each document $d \in \mathcal{D}$ is processed to extract a set of associated edges $\mathcal{G}_d \subseteq \mathcal{G}$. The full TKG is then defined as $\mathcal{G} = \cup_{d \in \mathcal{D}} \mathcal{G}_d$. In REEL, we introduce an additional step after TKG construction. That is, we pair each document $d \in \mathcal{D}$ with its extracted edges $\mathcal{G}_d \subseteq \mathcal{G}$ to form the document-edges pair set $P = \{(d, \mathcal{G}_d)\}$. This set is used to enable zero-shot TKGF generalization.

The strong performance of rule-based TKGF can be attributed to the logic rules used to reason about entities. Following TLogic (Liu et al., 2022), we construct such logic rules by first extracting temporal random walks for each relation $r \in \mathcal{R}_{in}$ in the TKG. These walks are then transformed into logic rules by replacing entities and timestamps with variables, with the first edge (bearing the latest timestamp) designated as the rule head and the remaining edges forming the body. Finally, we assess the confidence of each rule and discard those that do not meet a user-defined threshold to maintain the quality of our candidate rankings.

By following this procedure, we collect a set of high-quality rules for each seen relation $\mathcal{L}_r \forall r \in \mathcal{R}_{in}$. In REEL, these rules are applied not only to seen relations but also to the unseen relations as well. This is done by first identifying the highly similar seen relations, and then use those relations'

associated rules to perform predictions for the zero-shot link forecasting query.

4.2 Query Augmentation with LLMs

While TKGs represent real-world knowledge as structured graph information, LLMs represent knowledge as natural language text tokens (Devlin et al., 2019). While LLMs can accept a naive text representation of the test query, such as "(*subject, relation, ?, timestamp*)", this format is often too concise for the model to accurately interpret the quadruples within the context and structure of the TKG.

For instance, the quadruple (*Washington*, *propose*, *?*, *t*) can lead to ambiguity, as "Washington" may refer to either a person or a city which can affect the likely meaning of the relation "propose." To address this ambiguity, REEL translates the link forecasting query $(s, r, ?, t)$ into a more comprehensive text representation, incorporating relevant context and additional information that is omitted in a naive representation.

Following prior work (Ding et al., 2024), we use an LLM to enrich the relations with descriptions for more comprehensive semantics and to reduce ambiguity. In the previous example, the relation "propose" is ambiguous and could mean an offer of marriage or the submission of a plan. However, if the relation is supplemented with a description in the context of the query, such as "To propose is to suggest a plan for consideration," we reduce the ambiguity of the relation's meaning in the context of the query.

To further reduce ambiguity, we can optionally enrich the entity as well. To disambiguate the query entity s_q, we select some documents from the document-edges pair set that mention the entity, $P_{s_q} \subset \{d|(d, \mathcal{G}_d) \in P, (s, r, o, t) \in \mathcal{G}_d, s = s_q \text{ or } o = s_q\}$, then prompt the LLM to describe the entity using the documents as context. We can further refine the relation description by leveraging the LLM to define the query relation r_q in the context of the enriched query entity s_q.

With the descriptions in the conversation history, in the next turn of the conversation, the LLM is prompted to translate the link forecasting query into a natural language question q_{nl}. Intuitively, if the query entity "Washington" is determined to be a person, the interpretation of "propose" as in a marriage proposal should be higher than if "Washington" identifies a location.

The natural language translation of the original query, q_{nl}, reduces ambiguity but remains concise. To enrich the query and further improve retrieval performance, we take inspiration from HyDE (Gao et al., 2023) and generate a hypothetical document for the query. This document captures keywords and patterns that are likely to appear in responses to the query q_{nl} or similar query. To generate this hypothetical document, we provide the natural language query q_{nl} along with enriched relation and entity descriptions to an LLM. The model is then prompted to generate a response, which serves as the hypothetical document d_h (The English prompts used in this process are provided in Appendix D). Compared to using the query directly, the hypothetical document has more context and, as a result, increases the likelihood of selecting documents with relevant, similar relations.

4.3 Relevant Relation Retrieval For Unseen Relation

Although unseen relations do not appear in the training TKG, they are typically correlated with, or are similar to, relations that have appeared in the training TKG. Therefore, the rules that predict the relevant seen relations may also be predictive of the unseen relation. For example, although we may not have seen the relation "nominated for," we have seen the relevant relation "award" and can reason about "nominated for" using "award" as a proxy. Following this intuition, REEL reasons over a zero-shot query by first identifying the relevant seen relations, then reason about the unseen relation with the rules associated with these relevant seen relations.

To retrieve the relevant relations, we use the semantic similarity between the hypothetical document d_h and each document from the document-edges pair set. We call this similarity score the document similarity score s_d. Let $\mathbf{D}$ denote the embedding of document d (and $\mathbf{D_h}$ denote the embedding of document d_h). The similarity between d and d_h is the cosine similarity:

$$s_{d,d_h} = \frac{\mathbf{D}\mathbf{D_h}}{\|\mathbf{D}\|\|\mathbf{D_h}\|}.$$

Next, we construct P_{sel}, the set of the N document-edge pairs with the highest document similarity score, to identify the most relevant seen relations. Higher similarity score indicates a closer alignment with the hypothetical document representing the zero-shot query. This indicates the selected pairs are more likely to encapsulate the char-

acteristics and patterns found in the seen relations that are analogous to the unseen relation and thus is likely to contain relevant seen relations.

To identify these relations and their similarity to the unseen relation, we calculate the relation similarity score s for each relation $r \in \mathcal{R}_{P'_{sel}} = \{r|(d, \mathcal{G}'_d) \in P'_{sel}, (s, r, o, t) \in \mathcal{G}'_d\}$. First, we filter P_{sel}'s edges to only include relations that have rules associated with them to produce $P'_{sel} := \{(d, \mathcal{G}'_d)|(d, \mathcal{G}_d) \in \mathcal{P}_{sel}, \mathcal{G}'_d := \{(s, r, o, t)|(s, r, o, t) \in \mathcal{G}_d, \mathcal{L}(r) \neq \emptyset\}\}$. Then, we can compute the relation similarity score s_r:

$$s_r = \sum_{(d,\mathcal{G}'_d) \in P'_{sel}} \sum_{(s',r',o',t') \in \mathcal{G}'_d} \mathbb{I}_r(r') s_{d_h,d}$$

where $\mathbb{I}_r(r')$ is the indicator function generically defined as

$$\mathbb{I}_a(b) = \begin{cases} 1, & \text{if } a = b \\ 0, & \text{otherwise} \end{cases}.$$

The intuition behind the formulation of the relation similarity score s_r is that relations associated with the retrieved documents are likely to be analogous to the unseen relation. However, each document can have multiple associated relations, and all of those relations may not be relevant. Thus, we weigh the higher frequency relations higher because they are less likely to be noise.

4.4 Temporal Logic Rule-Based Candidate Entity Ranking

After finding similar seen relations, we can reason about the unseen relation r_q by applying the logic rules associated with the identified relevant seen relations $r \in \mathcal{R}_{P'_{sel}}$. Because each identified relation $r \in \mathcal{R}_{P'_{sel}}$ is considered to be similar to the unseen relation, we substitute the relation in the original query, generating a hypothetical link forecasting query for each relation: $(s_q, r, ?, t_q)$.

For each hypothetical link forecasting query, we apply the rule-based TKGF method to get the candidate entity ranking. We follow the procedure of TLogic[2] (Liu et al., 2022) and ground each rule $l \in \mathcal{L}_r$ associated with the query relation to the TKG to get a set of candidate object entities $e_c \in \mathcal{E}_r \subseteq \mathcal{E}$. Each candidate object entity e_c is then scored as a combination of rule confidence and timestamp difference:

$$f(l, e_c) = \alpha \cdot \text{conf}(l) + (1 - \alpha) \cdot \exp(-\lambda(t_q - t_1(l, e_c)))$$

where $\lambda > 0$, $\alpha \in [0, 1]$, and $t_1(l, e_c)$ denotes the earliest timestamp of the grounding of rule l at e_c.

After this process is performed for each rule associated with the relation, each entity $e \in \mathcal{E}$ has a potentially empty list of scores assigned by each rule. The aggregated score for the candidate entity for the seen relation r is:

$$C_r(e_c) = \begin{cases} 0, & \text{if } e_c \notin \mathcal{E}_r \\ 1 - \prod_{l \in \mathcal{L}_r}(1 - f(l, e_c)), & \text{otherwise.} \end{cases}$$

To generate the candidate entity ranking for the unseen relation r_q, we aggregate the hypothetical queries' candidate rankings weighted by the normalized relation similarity score defined as:

$$\hat{s_r} = \frac{s_r(r)}{\sum_{r' \in \mathcal{R}_{D'_{sel}}} s_r(r')}.$$

If a seen relation is more similar to the r_q, the candidate scores for that query using the seen relation is more likely to reflect the candidate scores for the query using the unseen relation and we should put more weight on those scores than those of seen relations which are less similar to r_q.

For each entity $e \in \{e|e \in \mathcal{E}, (d, \mathcal{G}'_d) \in D'_{sel}, (s, r, o, t) \in \mathcal{G}'_d, C^{\mathcal{L}}_r(e) \neq 0\}$, the candidate ranking score for the the unseen relation r_q is:

$$C^{\mathcal{L}}_{r_q}(e) = \sum_{r \in \mathcal{R}_{D'_{sel}}} \hat{s_r} C^{\mathcal{L}}_r(e).$$

Prior work has found that entities tend to correlate (Xu et al., 2023). To reflect this in our approach, we create an additional candidate ranking where, for each edge containing the subject entity of the link forecasting query s_q, every entity in the edge besides s_q are ranked according to their normalized frequency. Let $H = [o|(s, r, o, t) \in \mathcal{G}, s \neq s_q]$

$$C^{s_q}_{r_q}(e) = \frac{\sum h \in H(1 - \mathbb{I}_e(h))}{|H|}$$

Finally, to ensure all entities $\mathcal{E}$ are ranked, we create a third candidate ranking where entities are ranked according to their normalized frequency in the TKG. Let $A = [s|(s, r, o, t) \in$

[2]Although we describe an implementation which uses TLogic to generate rules and rank candidate entities, any rule-based approach that returns a set of candidate entities e_c and their associated probability scores $C^{\mathcal{L}}_r(e_c)$ can be used in hypothetical query candidate ranking generation.

		Unseen Relations				Seen Relations			
		MRR	H@1	H@3	H@10	MRR	H@1	H@3	H@10
zs-DuEE	**zrLLM**	0.121	0.063	0.083	0.264	0.506	0.382	0.563	0.793
	CENET	0.085	0.028	0.069	0.215	**0.583**	**0.430**	**0.686**	**0.916**
	TLogic	0.069	0.021	0.028	0.139	0.458	0.298	0.531	0.835
	Ours	**0.246**	**0.181**	**0.257**	**0.333**	0.458	0.298	0.531	0.835
zs-Wiki	**zrLLM**	0.136	0.073	0.175	**0.270**	0.455	0.370	0.487	0.641
	CENET	0.131	0.078	0.159	0.262	0.666	0.565	0.735	0.872
	TLogic	0.009	0.000	0.000	0.000	**0.773**	**0.700**	**0.844**	**0.878**
	Ours	**0.153**	**0.097**	**0.178**	0.254	**0.773**	**0.700**	**0.844**	**0.878**
zs-GDELT	**zrLLM**	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM
	CENET	0.157	0.034	0.156	0.456	0.304	**0.191**	0.317	0.553
	TLogic	0.193	0.034	**0.350**	0.499	**0.307**	0.166	**0.349**	**0.601**
	Ours	**0.250**	**0.141**	0.256	**0.525**	**0.307**	0.166	**0.349**	**0.601**

Table 1: Evaluation over the zero-shot datasets. Best results are bold, second-best results are underlined.

	Unseen Relations (MRR)			Seen Relations (MRR)		
	zs-DuEE	zs-Wiki	zs-GDELT	zs-DuEE	zs-Wiki	zs-GDELT
zrLLM	42.35%	3.82%	N/A	-13.20%	-31.68%	N/A
Ours	256.52%	1600.00%	29.53%	0.00%	0.00%	0.00%

Table 2: Comparison of zrLLM and our approach as change in MRR over baseline

$\mathcal{G}] \cup [o|(s, r, o, t) \in \mathcal{G}]$

$$C^{all}_{r_q}(e) = \frac{\sum a \in A(1 - \mathbb{I}_e(a))}{|H|}$$

Then, to perform candidate ranking, we order the three candidate rankings either as $C = [C^{\mathcal{L}}_{r_q}, C^{s_q}_{r_q}, C^{all}_{r_q}]$, or $[C^{s_q}_{r_q}, C^{\mathcal{L}}_{r_q}, C^{all}_{r_q}]$. The final candidate ranking for the link forecasting query with the unseen relationship is first the entities for which $C[0](e) \neq 0$ ordered by the score, then for $C[2]$, and finally for $C[1]$. We provide an analysis of the ranking order in Appendix E.

5 Experiments

5.1 Datasets

Our approach leverages both TKGs and textual descriptions. To our best knowledge, there are no existing datasets for the zero-shot temporal forecasting task that includes both a TKG, the documents used to create the TKG.

To fill in this gap, we curate three datasets for this setting: zs-DuEE, zs-Wiki, and zs-GDELT. We build upon prior work that has reformatted existing TKG datasets to be aligned with a set of documents (Han et al., 2023) and convert them to the zero-shot relational learning task by applying a similar procedure as in Ding et al. Details are available in Appendix A.

5.2 Baselines & Metrics

For the comparison of zero-shot TKGF performance, we evaluate the following methods:

1. **zrLLM:** This method is recent work that addresses our problem setting. We apply zrLLM to CENET, which represents the best combination from their evaluations (Ding et al., 2024).

2. **CENET:** This method is included to represent the baseline performance of embedding-based approaches on the zero-shot datasets we evaluate.

3. **TLogic:** TLogic (Liu et al., 2022) is a classic method upon which other rule-based approaches are based. Although there are other rule-based methods, TLogic was selected for its foundational significance.

To evaluate performance,we use the mean reciprocal rank (MRR), calculated on the average rank assigned to the correct entity candidate of a query, and Hits@k (for $k \in \{1, 3, 10\}$), the fraction of times the correct entity candidate is ranked within the top k of all candidates.

5.3 Main Results

We evaluate over each of the datasets with one run each in Table 1. Both zrLLM and our approach

	MRR	H@1	H@3	H@10
Llama3 70B	0.245	0.181	0.257	0.326
Llama3 8B	0.246	0.181	0.257	0.333

Table 3: Comparing the difference in the performance of our approach using Llama3 70B vs 8B on zs-DuEE

are able to reliably enhance the performance of the baseline model on unseen entities although REEL consistently outperforms zrLLM, achieving an average MRR improvement of 58.12% across all three datasets and having the highest MRR on the unseen relations across all three datasets. Both zrLLM and our approach are frameworks that enhance an existing TKGF model to improve performance on unseen relations. We compare the improvement in MRR over the unaugmented methods (CENET and TLogic respectively) in Table 2. While both methods demonstrate gains, REEL has a dramatic higher impact on scale of 1-2 orders of magnitude since, unlike embedding-based methods, rule-based approaches cannot to generalize to unseen relations and instead rely on naive ranking baselines. Furthermore, unlike zrLLM, REEL is able to guarantee performance preservation on seen relations by only applying the modified approach to unseen relations.

5.4 Ablation Studies

Table 4 shows the results of the ablations. Note that since we focus on the performance of the rule-generated candidate ranking, the table uses $[C^{\mathcal{L}}_{r_q}, C^{s_q}_{r_q}, C^{all}_{r_q}]$ ordering. Our ablations show that our method consistently has the best MRR.

Language model size. For the main results, we use Llama3 8B. In Table 3, compare the results of using Llama3 70B vs Llama3 8B are roughly equal (MRR of 0.245 versus 0.246 over the unseen relations) over the zs-DuEE dataset.

Relation Enrichment. The rows labeled "-EnrichRels" in Table 4 compares the performance of enriching the relation descriptions versus directly using the short relation names. For all datasets, this results in minor degradation in performance, indicating this is an important step for accurate query translation.

Query Translation. The rows labeled "-Translate" in Table 4 compares the performance of using a natural language query for hypothetical document generation versus the original query directly. Like the relation enrichment ablation, this results in minor degradation in performance, indicating this is an important step for hypothetical document generation.

		MRR	H@1	H@3	H@10
zs-DuEE	**Ours**	**0.242**	**0.167**	**0.243**	0.396
	-GenDoc	0.222	0.132	0.229	**0.410**
	-Translate	0.216	0.139	0.215	0.368
	-EnrichRels	0.219	0.160	0.208	0.333
zs-Wiki	**Ours**	**0.153**	**0.097**	**0.178**	**0.254**
	-GenDoc	0.139	0.085	0.154	0.225
	-Translate	0.147	0.096	0.169	0.225
	-EnrichRels	0.151	0.096	0.172	0.251
zs-GDELT	**Ours**	**0.208**	0.091	0.212	**0.471**
	-GenDoc	**0.208**	0.089	**0.218**	**0.471**
	-Translate	0.206	0.092	0.202	**0.471**
	-EnrichRels	**0.208**	**0.094**	0.209	0.463

Table 4: Comparing the performance of different variations of REEL

Hypothetical document. The rows labeled "-GenDoc" in Table 4 compares the performance of using a hypothetical document for document retrieval versus using the natural language query directly. For the zs-Wiki dataset, this severely degrades performance whereas for the other datasets, it does not affect the performance as much. This may indicate that for zs-Wiki, further enrichment is added through the hypothetical document generation whereas for the other two datasets, the query contains sufficient context.

6 Conclusion

We propose REEL, which offers a zero-shot generalization capability for rule-based temporal knowledge graph forecasting (TKGF), allowing such methods to rank candidate tail entities for unseen relations. The key insight of our approach is the use of a parallel corpus of documents to identify relevant seen relations to unseen relations and perform generalization based on those relations. We curate three datasets to evaluate generalization to unseen relations and compare performance to SOTA embedding-based TKGF techniques. Our experiments indicate that REEL achieves better performance than SOTA while also being capable of providing logical explanations.

7 Limitations

Some of the limitations that could be addressed in future work are: improving computational efficiency, improving the sophistication of the retrieval and inverse relation handling, and enhancing seen relations.

References

Kewei Cheng, Nesreen Ahmed, and Yizhou Sun. 2023. Neural compositional rule learning for knowledge graph reasoning. In *The Eleventh International Conference on Learning Representations*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

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, Mexico City, Mexico. Association for Computational Linguistics.

Zifeng Ding, Bailan He, Jingpei Wu, Yunpu Ma, Zhen Han, and Volker Tresp. 2023a. Learning meta-representations of one-shot relations for temporal knowledge graph link prediction. In *2023 International Joint Conference on Neural Networks (IJCNN)*, pages 1–10.

Zifeng Ding, Jingpei Wu, Zongyue Li, Yunpu Ma, and Volker Tresp. 2023b. Improving few-shot inductive learning on temporal knowledge graphs using confidence-augmented reinforcement learning. In *Machine Learning and Knowledge Discovery in Databases: Research Track: European Conference, ECML PKDD 2023, Turin, Italy, September 18–22, 2023, Proceedings, Part III*, page 550–566, Berlin, Heidelberg. Springer-Verlag.

Hao Gao, Yongqing Wang, Jiangli Shao, Huawei Shen, and Xueqi Cheng. 2022. User identity linkage across social networks with the enhancement of knowledge graph and time decay function. *Entropy*, 24(11).

Luyu Gao, Xueguang Ma, Jimmy Lin, and Jamie Callan. 2023. Precise zero-shot dense retrieval without relevance labels. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1762–1777, Toronto, Canada. Association for Computational Linguistics.

Zhen Han, Ruotong Liao, Jindong Gu, Yao Zhang, Zifeng Ding, Yujia Gu, Heinz Koeppl, Hinrich Schütze, and Volker Tresp. 2023. ECOLA: Enhancing temporal knowledge embeddings with contextualized language representations. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 5433–5447, Toronto, Canada. Association for Computational Linguistics.

Shibal Ibrahim, Wenyu Chen, Yada Zhu, Pin-Yu Chen, Yang Zhang, and Rahul Mazumder. 2022. Knowledge graph guided simultaneous forecasting and network learning for multivariate financial time series. In *Proceedings of the Third ACM International Conference on AI in Finance*, ICAIF '22, page 480–488, New York, NY, USA. Association for Computing Machinery.

Wei Jia, Ruizhe Ma, Weinan Niu, Li Yan, and Zongmin Ma. 2024. Sfte: Temporal knowledge graphs embedding for future interaction prediction. *Information Systems*, 125:102423.

Timotej Knez and Slavko Žitnik. 2023. Event-centric temporal knowledge graph construction: A survey. *Mathematics*, 11(23).

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc.

Xinyu Li, Fayuan Li, Lu Pan, Yuguang Chen, Weihua Peng, Quan Wang, Yajuan Lyu, and Yong Zhu. 2020. Duee: A large-scale dataset for chinese event extraction in real-world scenarios. In *Natural Language Processing and Chinese Computing: 9th CCF International Conference, NLPCC 2020, Zhengzhou, China, October 14–18, 2020, Proceedings, Part II*, page 534–545, Berlin, Heidelberg. Springer-Verlag.

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.

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, Mexico City, Mexico. Association for Computational Linguistics.

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*, 36(4):4120–4127.

Mehrnoosh Mirtaheri, Mohammad Rostami, Xiang Ren, Fred Morstatter, and Aram Galstyan. 2021. One-shot learning for temporal knowledge graphs. In *3rd Conference on Automated Knowledge Base Construction*.

Pouya Ghiasnezhad Omran, Kewen Wang, and Zhe Wang. 2021. An embedding-based approach to rule learning in knowledge graphs. *IEEE Transactions on Knowledge and Data Engineering*, 33(4):1348–1359.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473, Hong Kong, China. Association for Computational Linguistics.

Marco Postiglione, Daniel Bean, Zeljko Kraljevic, Richard JB Dobson, and Vincenzo Moscato. 2024. Predicting future disorders via temporal knowledge graphs and medical ontologies. *IEEE Journal of Biomedical and Health Informatics*, 28(7):4238–4248.

Pengda Qin, Xin Wang, Wenhu Chen, Chunyun Zhang, Weiran Xu, and William Yang Wang. 2020. Generative adversarial zero-shot relational learning for knowledge graphs. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8673–8680.

Meng Qu, Junkun Chen, Louis-Pascal Xhonneux, Yoshua Bengio, and Jian Tang. 2021. Rnnlogic: Learning logic rules for reasoning on knowledge graphs. In *International Conference on Learning Representations*.

Haseeb Shah, Johannes Villmow, Adrian Ulges, Ulrich Schwanecke, and Faisal Shafait. 2019. An open-world extension to knowledge graph completion models. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence and Thirty-First Innovative Applications of Artificial Intelligence Conference and Ninth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'19/IAAI'19/EAAI'19. AAAI Press.

Xiaozhi Wang, Tianyu Gao, Zhaocheng Zhu, Zhengyan Zhang, Zhiyuan Liu, Juanzi Li, and Jian Tang. 2021. KEPLER: A unified model for knowledge embedding and pre-trained language representation. *Transactions of the Association for Computational Linguistics*, 9:176–194.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. 2022. Emergent abilities of large language models. *Transactions on Machine Learning Research*. Survey Certification.

Siheng Xiong, Yuan Yang, Faramarz Fekri, and James Clayton Kerce. Tilp: Differentiable learning of temporal logical rules on knowledge graphs. In *The Eleventh International Conference on Learning Representations*.

Yi Xu, Junjie Ou, Hui Xu, and Luoyi Fu. 2023. Temporal knowledge graph reasoning with historical contrastive learning. In *AAAI*.

		zs-DuEE	zs-GDELT	zs-Wiki
Threshold		70	407	3822
Train	**Facts**	1246	76,380	211,787
	Rels	9	36	11
	Ents	196	147	9668
Val	**Facts**	309	35,349	35,049
	Rels	9	36	11
	Ents	153	138	4653
Test	**Facts**	144	4487	2534
	Rels	27	112	8
	Ents	84	95	862

Table 5: Dataset statistics

Zezhong Xu, Peng Ye, Hui Chen, Meng Zhao, Huajun Chen, and Wen Zhang. 2022. Ruleformer: Context-aware rule mining over knowledge graph. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 2551–2560, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Zefan Zeng, Qing Cheng, and Yuehang Si. 2023. Logical rule-based knowledge graph reasoning: A comprehensive survey. *Mathematics*, 11(21).

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 *AAAI*.

A Datasets Details

Following a similar set up as the one presented in prior zero-shot relational learning work (Ding et al., 2024) We identify the relations to set aside as the unseen relations by computing the frequencies of all relations in the training set and selecting as the zero-shot, unseen relations all relations with frequency below a set threshold. These relations are then removed from the training set and the test set is subset to only include the zero-shot relations from the evaluation split (the combination of the original test and validation sets) and the validation dataset to include only the seen relations.

We compute the frequency threshold for each dataset by selecting the threshold for which both the training and test have at least five relations and for which the test size is the largest. If there are multiple with the same test dataset size, we select the one with the bigger training dataset size.

The selected frequency threshold and dataset statistics are in Table 5. Our curated datasets are free for research use.

We briefly summarize the type of events in each of the datasets:

- **zs-DuEE:** the TKG generated from Chinese news articles covering social news such as product launches. The original creators of the dataset, Baidu Research (Li et al., 2020), state that they are "committed to provid[ing] these datasets at no cost for research and personal uses[3]."

- **zs-Wiki:** the TKG generated from Wikipedia articles focused on evolving knowledge such as affiliation and residence place information. The data comes from WikiData which has a license of CC0 1.0 Universal [4].

- **zs-GDELT:** a TKG generated from news reports and primarily covers social and political news. The website states the "entire GDELT database is 100% free and open[5]."

B Implementation Details

We use Pandas[6], Numpy[7], Transformers[8], PyTorch[9], and standard Python libraries. All experiments are run on a server with a Intel(R) Xeon(R) Gold 6240R CPU and four Tesla V100-SXM2-32GB GPUs. All baselines use the publicly-available source code provided by the authors except for the changes noted in Section 5.

For CENET and zrLLM+CENET, we use the best of oracle mode hard and soft for each dataset. For TLogic, we generate one set of rules with seed=12 and num_walks=200 and use the same rules for TLogic as well as REEL.

C Experimental Details

We note we modified TLogic so that in evaluation, rather than using the full ground truth TKG $\mathcal{G}_{gt}$, which includes quadruples from validation and test to rank the candidates, we use training TKG, $\mathcal{G}$. This is done to avoid potential data leakage since it is trivial to retrieve the correct object entity if the test query is already in the TKG.

For zrLLM, we follow their setting and use ChatGPT to generate the enriched relation descriptions.

For REEL, we use Llama3 8B as the LLM.

[3]https://ai.baidu.com/broad/introduction?dataset=duee
[4]https://www.wikidata.org/wiki/Wikidata:Main_Page
[5]https://www.gdeltproject.org/data.html#googlebigquery
[6]https://pandas.pydata.org
[7]https://numpy.org
[8]https://huggingface.co/docs/transformers/en/index
[9]https://pytorch.org

		MRR	H@1	H@3	H@10
zs-DuEE	**EE+ER**	0.242	0.167	0.243	0.396
	ER	0.225	0.146	0.236	0.340
zs-Wiki	**EE+ER**	0.151	0.096	0.170	0.251
	ER	0.153	0.097	0.178	0.254
zs-GDELT	**EE+ER**	0.250	0.141	0.256	0.525
	ER	0.250	0.141	0.256	0.525

Table 6: Comparison of using enriched entity and relation description versus just enriched relation description

D Prompts

Prompt: Subject Entity Enrichment

<document texts>
—
Given the above documents, please concisely tell me about the entity <subject entity>.

With the subject entity enrichment conversation turn in the conversation context, we prompt the LLM:

Prompt: Relation Enrichment

Please concisely describe the meaning of the relation <relation> in the context of <subject entity> being <relation> some other entity.

In the same conversation as above:

Prompt: Natural Language Translation

Given the context above, rewrite the following question in natural language. Give the rewritten version only and don't say anything extra. Do not add any quotes: "What is UNKNOWN ENTITY in this sentence: [<subject entity>] [<relation>] UNKNOWN_ENTITY?"

With the subject and relation enrichment in the conversation context:

Prompt: Hypothetical Document Generation

Given the context above, generate a paragraph to concisely answer the following question grounded in your factual knowledge: <generated natural language query translation of link forecasting query>"

E Hyperparameter Analysis

We presented REEL, a framework to generalize rule-based link forecasting methods to unseen relations. REEL includes an optional step for en-

		MRR	H@1	H@3	H@10
zs-DuEE	**RH**	0.242	0.167	0.243	0.396
	HR	0.246	0.181	0.257	0.333
zs-Wiki	**RH**	0.153	0.097	0.178	0.254
	HR	0.150	0.086	0.196	0.271
zs-GDELT	**RH**	0.208	0.091	0.212	0.471
	HR	0.250	0.141	0.256	0.525

Table 7: Comparison of using rule-first versus head entity correlation-first ranking ordering

riching entity descriptions and offers two ranking variations for evaluation.

Enriched Entity Descriptions. We compare the performance of including enriched entity descriptions and enriched relation descriptions in the context for hypothetical document generation (EE+ER) versus just including the enriched relation description (ER) in Table 6. We find that performance is mixed depending on the dataset with the extended context improving performance over the zs-DuEE dataset whereas not affecting performance for the zs-GDELT dataset and reducing performance for the zs-Wiki dataset. zs-DuEE is a dataset drawn from news articles whereas zs-Wiki is a much more diverse dataset. Since the context of the documents are much more similar for zs-DuEE, the additional context provided by enriching the entity using the documents may be more likely to be relevant compared to zs-Wiki's documents, and thus able to help generate an effective hypothetical document.

Ranking Ordering. REEL has two variations of ranking ordering: one where candidates ranked via rules is prioritized over the candidates ranked via subject entity correlation frequency distribution (RH; $[C^{\mathcal{L}}_{r_q}, C^{s_q}_{r_q}, C^{all}_{r_q}]$), and one where the ordering is switched (HR; $[C^{s_q}_{r_q}, C^{\mathcal{L}}_{r_q}, C^{all}_{r_q}]$). As indicated by the performance over the datasets in Table 7, there is not a strictly better ordering. For datasets where entities tend to occur together frequently, it may be better to rely on historic entity correlation whereas for datasets where the correlation is not as strong, such as zs-Wiki, it is better to rely on historic relation sequence correlation.

F Case Study

We provide a case study of zrLLM and our approach applied to a single test query to demonstrate how the results may be interpreted. In particular, we choose a test query from the zs-Wiki dataset: (*sopranos tv*, *nominated for*, *?*, *timestamp 65*).

Ours	zrLLM
(answer) primetime emmy award for outstanding lead actor - drama series	directors guild of america award for outstanding directing – drama series
screen actors guild award for outstanding actress - drama series	tca award for program of the year
tca award for individual achievement in drama	writers guild of america award for best screenplay – dramatic series
primetime emmy award for outstanding lead actress in a drama series	primetime emmy award for outstanding directing - drama series
list of golden globe awards: television, best series, drama	tca award for individual achievement in drama
tca award for outstanding new program	tca award for outstanding achievement in drama
directors guild of america award for outstanding directing – drama series	list of golden globe awards: television, best series, drama
(answer) golden globe award for best actress - television series (drama)	outstanding writing for a drama series
outstanding writing for a drama series	primetime emmy award for outstanding lead actress in a drama series
(answer) screen actors guild award for outstanding actress - drama series	(answer) screen actors guild award for outstanding actress - drama series

Table 8: Top ten candidates entities for examined query

There are four valid object entities for this query:

1. screen actors guild award for outstanding actress - drama series
2. primetime emmy award for outstanding lead actor - drama series
3. golden globe award for best actress - television series (drama)
4. primetime emmy award for outstanding guest actress - drama series

The top ten candidates entities for each method are listed in Table 8.

zrLLM. From the top ten candidate entities, all the entities are generally related to awards to the actor, actress, or director for their role in a drama series. However, the process to generate these outcomes is not transparent. Why one entity is ranked above another, and how to change the internal logic to change the ranking is inscrutable.

Ours. From the top ten candidate entities, three of the four correct completions to the test query. The first eight candidates (and two of the correct object entities) are selected via the rule-based approach and the last two items (including the expected answer for this specific query) are selected due to prior associations in the training KG.

This indicates the sequence of relations that generally lead to a "nominated for" relation between the subject entity and the object entity is missing and to boost the ranking of this answer in particular, more rules can be generated, either by rerunning the TKGF rule-based method or by a human with

domain-knowledge, that captures the sequence of events that had led to this connection or if the issue is that some edges are missing, those can be added into the TKG.

For the candidates that are selected via rules, one can gain further insight into their selection and relative ranking by looking at what rules were used and the support from the TKG. For this specific query, the seen relations used as a proxy for this unseen relation is just one: the relation "award received."

The relation "award received" has five rules associated with it, such as the rule which states if the show had previously received that award, it is likely to receive that award in the future, (e_1, *award received*, e_2) $\leftarrow$ (e_1, *award received,* e_2). For each of the rules, the top two entities have the same number of walks associated with them for each rule and as a result, have the same score. This indicates that the rules are not discriminative enough and if one wants performance to further improve, one may want to increase the number or quality of rules associated with the relation.

In both the case where rules are not used to make the prediction and where rules are used, the decision-making process is transparent and there are clear actions that can be taken to change the candidate rankings.