Question Answering Over Spatio-Temporal Knowledge Graph

Anonymous ACL submission

Abstract

Spatio-temporal knowledge graphs (STKGs) extend the concept of knowledge graphs (KGs) by incorporating time and location information. While the research community's focus on Knowledge Graph Question Answering (KGQA), the field of answering questions incorporating both spatio-temporal information based on STKGs remains largely unexplored. Furthermore, a lack of comprehensive datasets also has hindered progress in this area. To address this issue, we present STQAD, a dataset comprising 10,000 natural language questions for spatio-temporal knowledge graph question answering (STKGQA). Unfortunately, various state-of-the-art KGQA approaches fall far short of achieving satisfactory performance on our dataset. In response, we propose STCQA, a new spatio-temporal KGQA approach that utilizes a novel STKG embedding method named STComplEx. By extracting temporal and spatial information from a question, our QA model can better comprehend the question and retrieve accurate answers from the STKG. Through extensive experiments, we demonstrate the quality of our dataset and the effectiveness of our STKGQA method.

1 Introduction

In traditional KGs, facts are represented in the format of triplets, such as (*subject, relation, object*). Temporal knowledge graphs(TKGs) extend the KG by incorporating timestamps or time intervals (Gottschalk and Demidova, 2018), adding start and end time to the representation of facts, as (*subject, relation, object, start_time, end_time*). TKG promotes the research on temporal inference (Saxena et al., 2021; Mavromatis et al., 2022). In many scenarios, not only time information but also geographic information is an important attribute for describing entities or events (Hoffart et al., 2013),the integration of temporal and spatial information takes KGs to a new level (Zhang et al., 2021; Ji et al., 2023). Currently, STKG assist in many fields by providing a more precise representation of facts in both temporal and geographical dimensions (Wang et al., 2021; Chen et al., 2022a; Ge et al., 2022a,b). The fact in STKG is represented as (*subject, relation, object, start_time, end_time, geographic_location*).

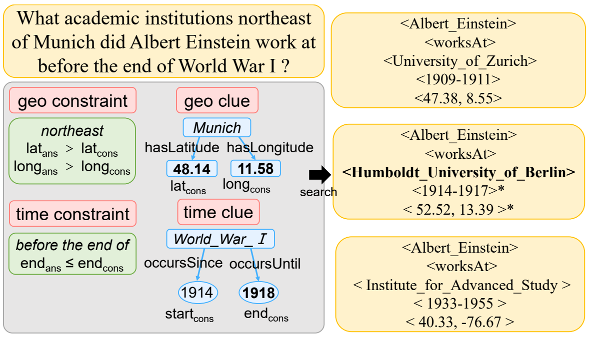

Figure 1: In STKGQA, it is necessary to extract spatio-temporal constraints and clues for answer retrieval. The spatio-temporal information marked with * satisfies the constraints of all clues in the question.

Currently, numerous works have already been conducted on temporal knowledge graph question answering (TKGQA) (Saxena et al., 2021; Mavromatis et al., 2022; Chen et al., 2022b; Shang et al., 2022), with a focus on extracting temporal information from natural language questions. Unfortunately, despite the long-standing availability of extending TKG to STKG, there is still a lack of further research on question answering using STKG.

Although some works have briefly discussed the methods of question answering (QA) based on STKG (Hoffart et al., 2013; Stringhini et al., 2019), these discussions only addressed questions involving either temporal or spatial features separately using expert-designed templates. They overlooked the situation where both time and space information occur in a single question. Additionally, the traditional query method uses the belonging rela-

Dataset	KG	Question Types			
		Multi-Entity	Multi-Relation	Temporal	Spatial
SimpleQuestions	FreeBase	×	×	×	×
GraphQ	FreeBase	×	✓	×	×
ComplexQuestions	FreeBase	×	✓	✓	×
QALD-7	DBpedia	×	×	×	×
LC-QuAD 2.0	Wikidata	×	×	×	×
TempQuestions	FreeBase	✓	✓	✓	×
CronQuestions	Wikidata	✓	✓	✓	×
STQAD(ours)	YAGO	✓	✓	✓	✓

Table 1: QA dataset comparison. There is currently no dataset that contains both temporal and spatial information in one question.

tion of locations for reasoning (eg, "*isCityOf*"), which is difficult to query the orientation or distance constraints in a question.

As shown in Figure 1, QA system needs to extract the potential temporal and geographical information from a question and search for the correct answers. From the STKG, we can obtain the geographical information related to "*Munich*" and the temporal information related to "*World War I*". Then we need to search all answers associated with the central entity "*Albert Einstein*" by considering spatio-temporal constraints. Unfortunately, as shown in Table 1, no further discussions have been conducted on this specific issue at present. Furthermore, there is a lack of large-scale datasets available for evaluating the QA task on STKG.

Motivated by the aforementioned situations, we create a spatio-temporal question answering dataset (STQAD) specifically designed for STKGQA, utilizing YAGO2(Hoffart et al., 2013) as the underlying STKG. In order to enhance the quality of the questions, we initially select entities as answers from spatio-temporal subgraphs, then we formulate queries and generate constraints that closely resemble real-world scenarios, incorporating them into the questions.

Many existing KGQA systems fail to consider situations where spatio-temporal constraints coexist within a question, leading to poor performance on our QA tasks. To address this limitation, we propose a spatio-temporal ComplEx embedding-based question answering (STCQA) method for effectively answering spatio-temporal questions. While the KG embedding method has demonstrated success in KGQA models (Saxena et al., 2020; Sun et al., 2020; Saxena et al., 2021), our approach extends the ComplEx (Trouillon et al., 2016) to STKG and introduces a new STKG embedding model called STComplEx. Additionally, we utilize the spatio-temporal constraint fragment of the question and implicit spatio-temporal clues to infer the final answers. The contributions of our work are summarized below:

1. We introduce the STQAD dataset, consisting of 10,000 questions specifically designed for the STKGQA task. To the best of our knowledge, this is the first comprehensive STKGQA dataset that encompasses spatial and temporal constraints in each question.

2. We propose the STCQA framework to address the STKGQA task. This method extends the normal KG embedding method to STKG, enabling the framework to integrate spatio-temporal constraints and KG embedding, and facilitating question answering.

3. Our experiments demonstrate the quality of our dataset and the effectiveness of our KGQA approach. Furthermore, although we have obtained promising initial results, our dataset still offers ample opportunities for enhancing the STKGQA.

2 Related Works

2.1 QA Datasets

Datasets are crucial for advancing KGQA models with robust generalization. SimpleQ (Bordes et al., 2015) explores multi-task and transfer learning's impact on simple QA. GraphQ (Su et al., 2016) enables fine-grained QA system analysis. ComplexQuestions (Bao et al., 2016) measures QA system quality on multi-constraint questions. QALD-7 (Usbeck et al., 2017) offers question-answer pairs for evaluating RDF and linked data QA systems. LC-QuAD 2.0 (Dubey et al., 2019) provides a dataset to study converting natural language questions into formal queries.

To address temporal reasoning, TORQUE (Ning et al., 2020) poses multiple-choice temporal questions with context. TempQuestions (Jia et al., 2018) defines temporal questions using trigger words to filter out irrelevant QA datasets. CronQuestions (Saxena et al., 2021) is a large TKGQA dataset with temporal KG and natural language questions. However, there's no large-scale evaluation dataset for the STKGQA task, leading most spatial QA frameworks to rely on geographic information systems (GIS) for geographic analysis.

2.2 KGQA

Numerous studies have explored KGQA. Query graph extraction has been proposed to address KBQA task (Yih et al., 2015; Bao et al., 2016). Embedding-based methods have also been proven to solve the KBQA (Dai et al., 2016; Hao et al., 2017; Lukovnikov et al., 2017; Févry et al., 2020). Neural network methods rely on learning a scoring function to rank candidate answers (Dai et al., 2016; Hao et al., 2017; Lukovnikov et al., 2017; Févry et al., 2020).

Temporal graph representation-based QA approaches (Saxena et al., 2021; Mavromatis et al., 2022; Chen et al., 2022b) use the TKG embedding method to learn entity, relation, and timestamp embeddings. The answer is determined by scoring the distance between the TKG embedding and the question embedding.

In previous research on STKGQA, studies have explored spatio-temporal question answering systems that utilize RDF KGs (Stringhini et al., 2019; Yin et al., 2019). Additionally, certain studies have focused on domain-specific STKGQA and have employed rule-based approaches for answer retrieval in STKGs (Del Mondo et al., 2021; Dopler and Scholz, 2021). However, these methods mainly present frameworks and lack extensive evaluations on large-scale datasets.

3 Dataset Generation

This section is divided into two parts: initially, we expand a TKG embedding dataset (García-Durán et al., 2018) to the spatial dimension, creating a dataset for our embedding model; then we generate a dataset for STKGQA leveraging this dataset along with STKG information.

3.1 STKG Embedding Dataset Generation

YAGO15k (García-Durán et al., 2018) serves as a TKG embedding dataset, it is based on FREEBASE15K (Bordes et al., 2013) (FB15K), with entities aligned to FB15K using the *sameAs* relation from YAGO2 (Hoffart et al., 2013). To enrich the spatial information in YAGO15K, we identify the "*happenedIn*" relation, which signifies the location of the facts. However, some KG facts lack geographical information, we utilize the coordinate information of the object and determine its relation type to supplement the facts' locations. For instance, consider the fact (*Albert_Einstein, worksAt, Humboldt_University_of_Berlin, 1914, 1917*), which lacks a "*happenedIn*" relation in YAGO2 for location labeling. We can assign the location coordinates of "*Humboldt_University_of_Berlin*" as $\langle 52.52, 13.39\rangle$ to complete the fact.

Table 2: Statistics of the STKG embedding dataset

Statistics type	Quantity
Entities	15,403
Relations	113
Facts	138,056
Distinct timestamp	572
Distinct location	2,235
Time span	2-3150
Train	110,851 [33,799]
Validation	13,858 [4,165]
Test	13,853 [4,221]

The storage format of a complete spatio-temporal fact is (*subject, relation, object, occursSince, start_time, occursIn, coordinate*) and (*subject, relation, object, occursUntil, end_time, occursIn, coordinate*). To provide a more accurate representation, we add spatio-temporal annotation to each relation. For instance, we use "*worksAt_occursSince_occursIn*" and "*worksAt_occursUntil_occursIn*" to denote the origin relation "*worksAt*". Table 2 shows the statistical overview of our dataset, where the facts with spatio-temporal information are in brackets.

3.2 STKGQA Dataset Generation

Based on the STKG embedding dataset, we utilize its facts to create a QA dataset. We specifically choose facts that encompass spatio-temporal details as potential candidates. We first generate the spatio-temporal constraints and clues required in the question, and then integrate these elements to generate the final question.

3.2.1 Spatio-temporal Constraint and Clue Generation

To answer natural language questions using STKG, we must begin with the question's central entity and utilize spatio-temporal constraints to locate answers on the KG. Therefore, question generation can be seen as the reverse of question answering, where the answer is used to derive all constraints and clues of the question. Furthermore, there should be a stronger correlation between entities and spatio-temporal constraints in the question, which would make the question more closely aligned with real-world scenarios.

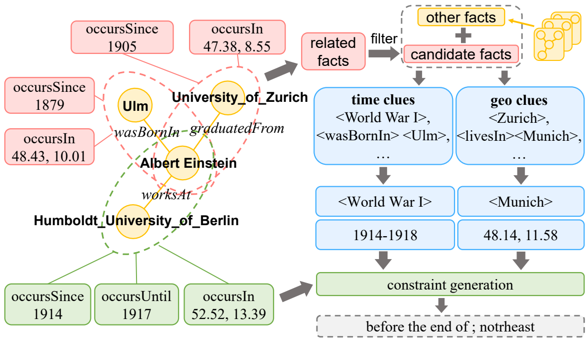

Figure 2: An example of constraint and clue generation.

The process of generating question constraints is shown in Figure 2. We utilize the fact (*Albert_Einstein, worksAt, Humboldt_University_of_Berlin, occursSince, 1914, occursUntil, 1917, occursIn, 52.52, 13.39*) as an answer fact, with *Albert_Einstein* being the central entity and *Humboldt_University_of_Berlin* as the answer.

To facilitate the generation of spatio-temporal constraints, we search for facts related to *Albert_Einstein* that contain both time and location information. Examples of such facts include (*Albert_Einstein, wasBornIn, Ulm, occursSince, 1879, occursIn, 48.43, 10.01*) and (*Albert_Einstein, graduatedFrom, University_of_Zurich, occursSince, 1905, occursIn, 47.38, 8.55*). We consider these facts as highly relevant candidates alongside the central entity. Some KG relations, such as "*influence*" and "*linksTo*", are ambiguous and challenging to convert into constraints, so we filter them out. Consequently, we obtain a set of high-quality candidate facts associated with the central entity. However, this strict approach may result in an insufficient number of candidate facts. To address this issue, we conduct a KG search involving entities, times, and places related to the central entity "*Albert_Einstein*", thereby acquiring additional facts to complement our fact set.

Table 3: Statistics of the STQAD.

	Train	Dev	Test
Single Timestamp Constraint	7729	970	978
Double Timestamp Constraint	271	30	22
Single Direction Constraint	4711	597	603
Double Direction Constraint	3289	403	397
Average Sentence Length	16.34	16.33	16.38
Overall Number	8000	1000	1000

Our fact set is divided into two subsets: the candidate time clue set and the candidate location clue set. We randomly select one clue from each of these two clue sets to generate the constraints of the question. By comparing the spatio-temporal clues with the answer fact, we derive constraints such as the time constraint "*before the end of*" and the location constraint "*northeast*" as illustrated in Figure 2.

3.2.2 Question Generation

After obtaining all necessary elements for sentence generation, we input the central entities relations, and spatio-temporal constraints into sentence templates to generate questions. To ensure variety, we have devised multiple templates for each fact, with several possible replacements for words within the templates. For example, in the question "*What {academic institutions} {northeast of} {Munich} did {Albert Einstein} work at {before the end of} {World War I}?*", the term *academic institutions* in the template can be replaced with *academies* or *academic organizations*. Appendix A.3 shows the specific format of the generated question. Additionally, the questions generated by the template are paraphrased using the ChatGPT (Wu et al., 2023). In this way, we expand the diversity of sentences and also ensures the quality of our dataset. Finally, we construct the STQAD containing 10,000 complex questions.

Table 3 presents the statistical information of the dataset. The "single time constraint" refers to the limitations imposed by a single timestamp, such as "before" and "after", the "double time constraint" pertains to limitations that must fall within a time interval, such as "during". It is worth noting that due to the small number of time interval type facts in YAGO15k and our strict constraint generation rules, the total number of "double time constraint" type questions in our dataset is limited. The "single direction constraint" involves a restriction to a single direction, such as "east", whereas the "double direction constraint" involves a constraint that encompasses two directions, such as "northeast".

4 STKGQA method

4.1 Overview

In this section, we initially introduce an embedding method called STComplex to the STKG. Building upon this method, we further develop a QA framework named STCQA. The architecture of the framework is depicted in Figure 3.

4.2 STKG Embedding

A STKG $\mathcal{K} := (\mathcal{E}, \mathcal{R}, \mathcal{T}, \mathcal{L}, \mathcal{F})$ contains a set of entities $\mathcal{E}$, a set of relations $\mathcal{R}$, a set of timestamps $\mathcal{T}$, a set of locations $\mathcal{L}$, and a set of facts $\mathcal{F}$. Each fact $(s, r, o, t, l) \in \mathcal{F}$ is a tuple where $s, o \in \mathcal{E}$ denote the subject and object entities, respectively, $r \in \mathcal{R}$ denotes the relation between them, $t \in \mathcal{T}$ is the timestamp of the fact, and $l \in \mathcal{L}$ is the location of the fact.

ComplEx (Trouillon et al., 2016) represents each entity e as a complex vector $\mathbf{e}$. Each relation r is represented as a complex vector $\mathbf{r}$ as well. The score ϕ_C of a fact (s, r, o) is

$$\phi_C\left(\mathbf{e}_s, \mathbf{r}, \overline{\mathbf{e}}_o\right) = \operatorname{Re}\left(\left\langle \mathbf{e}_s, \mathbf{r}, \overline{\mathbf{e}}_o \right\rangle\right)$$

where $\operatorname{Re}(.)$ denotes the real part, $\overline{(.)}$ is the complex conjugate of the embedding vector.

TComplEx (Lacroix et al., 2020) is an extension of the ComplEx KG embedding method designed for TKGs. TComplEx represents each timestamp t as a complex vector $\mathbf{t}$ and the score ϕ_T of a fact (s, r, o, t) is

$$\phi_T\left(\mathbf{e}_s, \mathbf{r}, \overline{\mathbf{e}}_o, \mathbf{t}\right) = \operatorname{Re}\left(\left\langle \mathbf{e}_s, \mathbf{r} \odot \mathbf{t}, \overline{\mathbf{e}}_o \right\rangle\right)$$

where $\odot$ is the element-wise product.

In this work, we propose an extension of TComplEx that incorporates location information. We introduce STComplEx, where the location coordinate l is represented as a complex vector $\mathbf{l}$. The scoring function for STComplEx is defined as

$$\phi_{ST}\left(\mathbf{e}_s, \mathbf{r}, \overline{\mathbf{e}}_o, \mathbf{t}\right) = \operatorname{Re}\left(\left\langle \mathbf{e}_s, \mathbf{r} \odot \mathbf{t} \odot \mathbf{l}, \overline{\mathbf{e}}_o \right\rangle\right)$$

All embedding vectors are learned such that the scoring function ϕ_{ST} assigns higher scores to valid facts $(s, r, o, t, l) \in \mathcal{F}$ compared to invalid facts $(s', r', o', t', l') \notin \mathcal{F}$. Formally, we have

$$\phi_{ST}\left(\mathbf{e}_s, \mathbf{r}, \overline{\mathbf{e}}_o, \mathbf{t}, \mathbf{l}\right) > \phi_{ST}\left(\mathbf{e}_s', \mathbf{r}', \overline{\mathbf{e}}_o', \mathbf{t}', \mathbf{l}'\right)$$

The embedding learning procedure employed by STComplEx enables the inference of missing facts such as $(s, r, ?, t, l)$ over an incomplete STKG.

4.3 Constraint Fragment Generation

The incorporation of spatio-temporal constraints is crucial for STKGQA. To capture these constraints, we employ a keyword-based approach. Initially, we identify specific spatio-temporal keywords within a given question, such as "after" and "northeast." We then extract the phrases situated between these spatial or temporal keywords and the associated entities, utilizing them as fragments for constraint modeling. To differentiate between the two types of constraints, we utilize distinctive tokens: [TC] for timestamp constraints and [GC] for geo constraints. Figure 3 illustrates this process, where the combined phrase "[TC] before the end of [GC] northeast" is encoded as v_p using BERT (Devlin et al., 2018). This vector enhances the spatio-temporal representation of the question, enabling improved answer retrieval by the system.

4.4 Entity Type Annotation

Entities have three types in a question involving spatio-temporal information: regular entities, entities implying time information, and entities implying geographic information. The determination of entity type relies on constraint keywords, such as "*after*" and "*northeast*", which appear before the entity. To label different types of entities in the original question, we use special tokens [ENT], [TS], and [GEO]. These labels serve the purpose of mitigating the impact of different entities on sentence meaning, enabling the encoder to focus solely on the semantics of the question. Each token in the sentence is encoded by BERT into a vector space. Figure 3 illustrates the masking of the original question as "*What academic institutions northeast of* [GEO] *did* [ENT] *work at before the end of* [TS]?"

4.5 Question Embedding Generation

The generation of question embeddings involves three main stages. Firstly, relevant entities are substituted with their corresponding implicit information based on spatio-temporal constraints and special tokens. For instance, in Figure 3, the implicit time information "*1918*" for the entity "*World War I*" is searched in the STKG using "[TS]" and "*before the end of*". When an entity possesses implicit time information, its time value denoted as t is

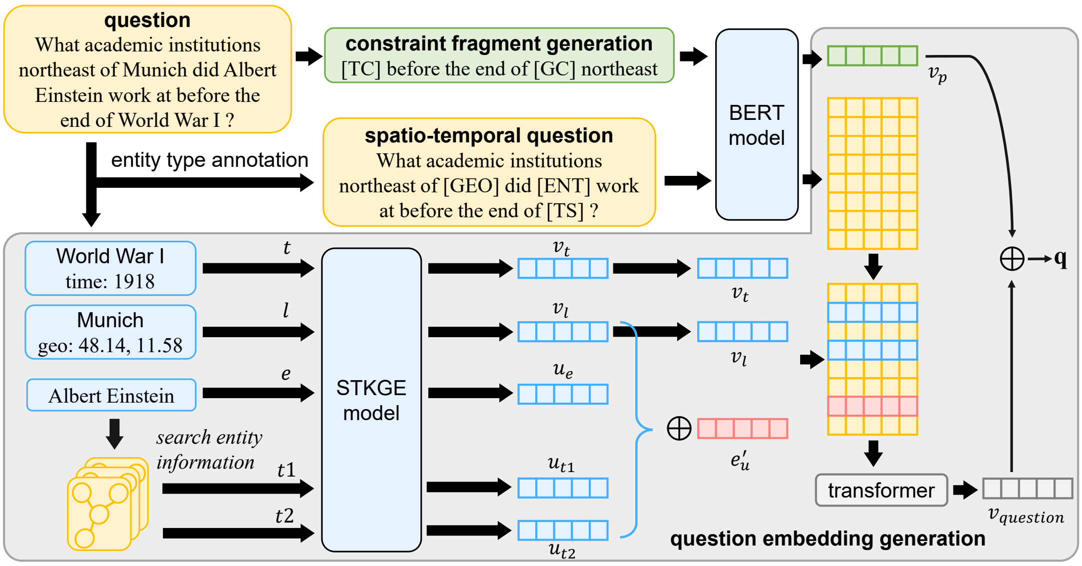

Figure 3: Our framework comprises three modules: constraint fragment generation, entity type annotation, and question embedding generation. (i) Constraint fragment generation: identifies spatio-temporal constraints within the question and encodes relevant clues. (ii) Entity type annotation: identifies entity types inferred from the spatio-temporal constraints in the question. (iii) Question embedding generation module: integrate entity representation and spatio-temporal embedding in STKG into question vector q.

encoded into a vector representation v_t using the STComplEx model. Similarly, for entities with implicit geographic location, the entity's geographic coordinates denoted as l are encoded into a vector representation v_l. For central entities denoted as e, all facts related to the entity in the STKG are retrieved, and the earliest time $t1$ and the latest time $t2$ in all facts are selected as the time dimension representation. We combine the temporal representation u_{t1} and u_{t2}, the spatial representation v_l, and the central entity representation u_e to form an entity representation u'_e that incorporates spatio-temporal information.

Next, we replace the embedding of special tokens with the STKG embedding of the entities respectively, considering their positions within the entity and token. For example, the special token [GEO] is replaced with the embedding representation v_l of the geographical coordinates *48.14, 11.58* in STKG.

Finally, after replacing all embeddings of special tokens, an information fusion layer is utilized. This layer incorporates a dedicated learnable encoder called $Transformer(.)$, which consists of two transformer encoding layers (Vaswani et al., 2017). The encoder enables the tokens in the question to attend to each other, thus effectively combining context, entity, location-aware information, and time-aware information. The question representation vector, denoted as $v_{question}$, is then combined with the spatio-temporal constraint fragment representation, denoted as v_p, to obtain the final question representation, denoted as q.

4.6 Answer Prediction

We employ the scoring function based on the STComplEx model to determine the final answer. The final score of an entity $\epsilon \in \mathcal{E}$ being the answer is given by

$$\max\left(\phi_{ST}\left(\mathbf{u}_e, \mathbf{W}_E\mathbf{q}, \mathbf{e}_\epsilon, \mathbf{v}_t, \mathbf{v}_l\right)\right)$$

where $\mathbf{u}_e$, $\mathbf{v}_t$, $\mathbf{v}_l$ are the annotated general entity, timestamp and location. We regard the semantic representation of the question as a relation, and $\mathbf{W}_E$ is a $D \times D$ learnable matrix specific for answer prediction.

5 Experiments

5.1 Experimental Setup

We evaluate our model using hit ratios (HR) as the primary evaluation metric. Additionally, we utilize the mean reciprocal rank (MRR) to evaluate the effectiveness of the STKG embedding. We assess several KGQA models on STQAD, using their performance as baselines. Appendix A.1 introduces the implementation of training and the setting of hyper-parameters.

5.2 Main Results and Analysis

5.2.1 STKG Embedding Results

The results of the STKG embeddings are presented in Table 4. Our model demonstrates a significant enhancement in the HR compared to existing embedding models. This emphasizes the indispensability of incorporating spatio-temporal information into the scoring function for STKG embedding. Notably, the disparity between the ComplEx (Trouillon et al., 2016) and TComplEx (Lacroix et al., 2020) is negligible, suggesting that the mere addition of temporal knowledge does not effectively improve the HR in the STKG embedding method. This observation emphasizes the crucial role of spatial relations in the context of STKG.

Table 4: Results for STKG embedding.

	Hit@1	Hit@3	Hit@10	MRR
ComplEx	21.72	34.30	49.96	30.80
TComplEx	21.68	34.75	50.15	31.01
STComplEx	**40.71**	**48.45**	**58.41**	**46.56**

5.2.2 STKGQA Results

The results of STKGQA are presented in Table 5. Our findings indicate that KG embedding-augmented methods outperform those based on large pre-trained LMs. Our experiments demonstrate that incorporating specific STKG embeddings enhances the model's ability to capture temporal and spatial clues.

Table 5: Results for STKGQA.

	Hit@1	Hit@3	Hit@10
BERT	39.10	53.70	66.87
RoBERTa	32.37	46.77	60.83
EmbedKGQA	38.00	53.03	56.00
CronKGQA	34.60	49.33	62.17
TempoQR	54.97	66.03	76.23
SubGTR	58.23	68.73	78.50
STCQA	**61.63**	**76.67**	**84.17**

By incorporating spatio-temporal information and using the STComplEx embedding model, our method outperforms traditional embedding-based QA approaches. Transformer-based models for STKGQA task typically necessitate a substantial and varied dataset for successful training, making them more costly compared to our approach. EmbedKGQA (Hu et al., 2017), which employs the ComplEx (Trouillon et al., 2016) embedding model, lacks the ability to perform inference on STKG. Due to the poor performance of the TComplEx (Lacroix et al., 2020) embedding model on our dataset, the QA methods CronKGQA (Saxena et al., 2021), TempoQR (Mavromatis et al., 2022), and SubGTR (Chen et al., 2022b), which rely on this embedding model, fell short of achieving the expected results. Although SubGTR incorporates a subgraph reasoning module that effectively handles complex zero-shot questions, it still requires enhancements in location inference capabilities for our STKGQA datasets. Appendix A.2 provides an introduction to the implementations of all baselines. Our model significantly surpasses existing baselines, establishing itself as the state-of-the-art solution.

5.3 Ablation Study

Table 6 presents the results of the ablation experiments, which investigate the contributions of each module in STCQA. We compare the performance of the following settings:

w/o constraint fragment generation Instead of utilizing the constraint fragment generation module, we use the transformer output as the final question representation.

w/o entity annotation The entity type is not annotated, and the spatio-temporal clues do not replace the entity token in the question. We solely employ the embedding model to encode and substitute the entities in the question.

w/o search entity information The final entity representation solely relies on the original encoded representation of the entity in the embedding model, and does not incorporate time and location embeddings in STKG.

Table 6: Ablation study for STCQA.

	Hit@1	Hit@3	Hit@10
STCQA	**61.63**	**76.67**	**84.17**
w/o prompt generation	60.47	74.93	83.27
w/o entity annotation	59.73	73.73	83.10
w/o search entity information	57.97	71.83	81.30

5.4 Effect of Dataset Size

Larger datasets typically offer a greater number of training samples, a crucial factor in training accurate and generalizable models. Figure 4 illustrates

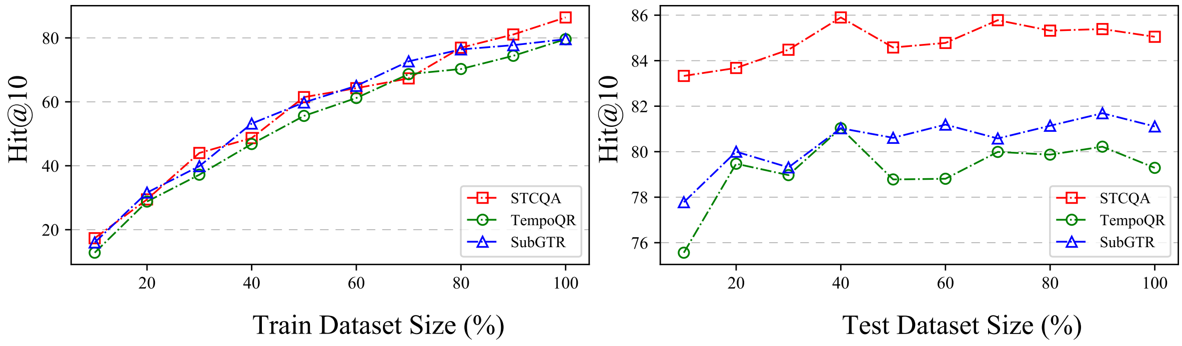

Figure 4: Model performance (hits@10) vs. dataset size (percentage) for STCQA, TempoQR and SubGTR.

the impact of dataset size on model performance. Increasing the training dataset size from 10% to 100% leads to a steady improvement in model performance. This trend remains consistent across different models, validating the effectiveness of a large-scale dataset for training STKGQA models.

Similarly, a larger test dataset can yield more reliable and statistically significant evaluation results. With a fixed training dataset size, we progressively enlarge the validation set and test set from 10% to 100%. The results obtained from the test dataset demonstrate that employing a larger amount of evaluation data is advantageous for ascertaining a more stable performance across various QA methods implemented on STQAD.

5.5 Effect of Spatio-temporal Constraint Relevance

The correlation between spatio-temporal constraints and central entities is a significant indicator for evaluating dataset quality. Using irrelevant entities in the question does not reflect real-world scenarios. For instance, consider the following questions: "*Which university in the northeast of Munich did Einstein work at after World War I?*" and "*Which university in the northwest of Beijing did Einstein work at after the 25th Academy Awards?*" Clearly, the former question is more representative of the real situation because the entities mentioned in it are more closely related to the central entity (as discussed in Section 3.2.1).

Table 7: Model performance on different spatio-temporal constraint relevance dataset.

	30% STCQA Dataset	Low Spatio-temporal Relevance Dataset
STCQA	**44.00**	32.67
TempoQR	**37.20**	30.93
SubGTR	**39.87**	30.80

We replaced the original facts in the question with facts that satisfy the constraints but have low relevance. This process allowed us to construct a dataset consisting of low co-occurrence entities (such as *Einstein* and *the 25th Academy Awards*). Due to the constraint on the number of facts, the size of the dataset is 30% of the STQAD dataset. For the sake of fairness, we used equally sized datasets for evaluation. As indicated in Table 7, the model performs better on datasets with a higher spatio-temporal constraint correlations, thereby validating our hypothesis.

6 Conclusion

In this paper, we introduce STQAD, a new dataset for STKGQA. While there have been numerous KGs based on spatio-temporal information, the existing KGQA datasets lack discussions regarding scenarios involving spatio-temporal information for reasoning. To the best of our knowledge, STQAD is the first dataset that comprises a substantial number of questions that necessitate both temporal and spatial inference. To ensure the realistic situation of the questions in the dataset, we employ strict constraints during question generation. The availability of large datasets not only enables model evaluation but also presents an opportunity for model training. Through experiments, we demonstrate that the performance of certain methods on the STKGQA task steadily improves with an increase in the training dataset size. Additionally, we propose a novel method, STCQA, which leverages STKG embeddings for the QA task. In our experiments, STCQA outperforms all baseline methods. These results indicate that STKG embeddings can be effectively utilized for STKGQA, although there is still considerable room for improvement.

References

Junwei Bao, Nan Duan, Zhao Yan, Ming Zhou, and Tiejun Zhao. 2016. Constraint-based question answering with knowledge graph. In *Proceedings of COLING 2016, the 26th international conference on computational linguistics: technical papers*, pages 2503–2514.

Antoine Bordes, Nicolas Usunier, Sumit Chopra, and Jason Weston. 2015. Large-scale simple question answering with memory networks. *arXiv preprint arXiv:1506.02075*.

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.

Wei Chen, Huaiyu Wan, Shengnan Guo, Haoyu Huang, Shaojie Zheng, Jiamu Li, Shuohao Lin, and Youfang Lin. 2022a. Building and exploiting spatial–temporal knowledge graph for next poi recommendation. *Knowledge-Based Systems*, 258:109951.

Ziyang Chen, Xiang Zhao, Jinzhi Liao, Xinyi Li, and Evangelos Kanoulas. 2022b. Temporal knowledge graph question answering via subgraph reasoning. *Knowledge-Based Systems*, page 109134.

Zihang Dai, Lei Li, and Wei Xu. 2016. Cfo: Conditional focused neural question answering with large-scale knowledge bases. *arXiv preprint arXiv:1606.01994*.

Géraldine Del Mondo, Peng Peng, Jérôme Gensel, Christophe Claramunt, and Feng Lu. 2021. Leveraging spatio-temporal graphs and knowledge graphs: Perspectives in the field of maritime transportation. *ISPRS International Journal of Geo-Information*, 10(8):541.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Silvia Dopler and Johannes Scholz. 2021. A prototypical geospatial knowledge graph and spatio-temporal question answering for supply chain visibility.

Mohnish Dubey, Debayan Banerjee, Abdelrahman Abdelkawi, and Jens Lehmann. 2019. Lc-quad 2.0: A large dataset for complex question answering over wikidata and dbpedia. In *The Semantic Web–ISWC 2019: 18th International Semantic Web Conference, Auckland, New Zealand, October 26–30, 2019, Proceedings, Part II 18*, pages 69–78. Springer.

John Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7).

Thibault Févry, Livio Baldini Soares, Nicholas FitzGerald, Eunsol Choi, and Tom Kwiatkowski. 2020. Entities as experts: Sparse memory access with entity supervision. *arXiv preprint arXiv:2004.07202*.

Alberto García-Durán, Sebastijan Dumančić, and Mathias Niepert. 2018. Learning sequence encoders for temporal knowledge graph completion. *arXiv preprint arXiv:1809.03202*.

Xingtong Ge, Yi Yang, Jiahui Chen, Weichao Li, Zhisheng Huang, Wenyue Zhang, and Ling Peng. 2022a. Disaster prediction knowledge graph based on multi-source spatio-temporal information. *Remote Sensing*, 14(5):1214.

Xingtong Ge, Yi Yang, Ling Peng, Luanjie Chen, Weichao Li, Wenyue Zhang, and Jiahui Chen. 2022b. Spatio-temporal knowledge graph based forest fire prediction with multi source heterogeneous data. *Remote Sensing*, 14(14):3496.

Simon Gottschalk and Elena Demidova. 2018. Eventkg: A multilingual event-centric temporal knowledge graph. In *The Semantic Web: 15th International Conference, ESWC 2018, Heraklion, Crete, Greece, June 3–7, 2018, Proceedings 15*, pages 272–287. Springer.

Yanchao Hao, Yuanzhe Zhang, Kang Liu, Shizhu He, Zhanyi Liu, Hua Wu, and Jun Zhao. 2017. An end-to-end model for question answering over knowledge base with cross-attention combining global knowledge. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 221–231.

Johannes Hoffart, Fabian M Suchanek, Klaus Berberich, and Gerhard Weikum. 2013. Yago2: A spatially and temporally enhanced knowledge base from wikipedia. *Artificial intelligence*, 194:28–61.

Sen Hu, Lei Zou, Jeffrey Xu Yu, Haixun Wang, and Dongyan Zhao. 2017. Answering natural language questions by subgraph matching over knowledge graphs. *IEEE Transactions on Knowledge and Data Engineering*, 30(5):824–837.

Hao Ji, Li Yan, and Zongmin Ma. 2023. Fstre: Fuzzy spatiotemporal rdf knowledge graph embedding using uncertain dynamic vector projection and rotation. *IEEE Transactions on Fuzzy Systems*.

Zhen Jia, Abdalghani Abujabal, Rishiraj Saha Roy, Jannik Strötgen, and Gerhard Weikum. 2018. Tempquestions: A benchmark for temporal question answering. In *Companion Proceedings of the The Web Conference 2018*, pages 1057–1062.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Timothée Lacroix, Guillaume Obozinski, and Nicolas Usunier. 2020. Tensor decompositions for temporal knowledge base completion. *arXiv preprint arXiv:2004.04926*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019.

Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Denis Lukovnikov, Asja Fischer, Jens Lehmann, and Sören Auer. 2017. Neural network-based question answering over knowledge graphs on word and character level. In *Proceedings of the 26th international conference on World Wide Web*, pages 1211–1220.

Costas Mavromatis, Prasanna Lakkur Subramanyam, Vassilis N Ioannidis, Adesoji Adeshina, Phillip R Howard, Tetiana Grinberg, Nagib Hakim, and George Karypis. 2022. Tempoqr: temporal question reasoning over knowledge graphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 5825–5833.

Qiang Ning, Hao Wu, Rujun Han, Nanyun Peng, Matt Gardner, and Dan Roth. 2020. Torque: A reading comprehension dataset of temporal ordering questions. *arXiv preprint arXiv:2005.00242*.

Apoorv Saxena, Soumen Chakrabarti, and Partha Talukdar. 2021. Question answering over temporal knowledge graphs. *arXiv preprint arXiv:2106.01515*.

Apoorv Saxena, Aditay Tripathi, and Partha Talukdar. 2020. Improving multi-hop question answering over knowledge graphs using knowledge base embeddings. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 4498–4507.

Chao Shang, Guangtao Wang, Peng Qi, and Jing Huang. 2022. Improving time sensitivity for question answering over temporal knowledge graphs. *arXiv preprint arXiv:2203.00255*.

Pedro Henrique Ferreira Stringhini, Julio Cesar dos Reis, Relatório Técnico-IC-PFG, and Projeto Final de Graduação. 2019. Spatio-temporal question answering based on rdf knowledge bases.

Yu Su, Huan Sun, Brian Sadler, Mudhakar Srivatsa, Izzeddin Gür, Zenghui Yan, and Xifeng Yan. 2016. On generating characteristic-rich question sets for qa evaluation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 562–572.

Haitian Sun, Andrew Arnold, Tania Bedrax Weiss, Fernando Pereira, and William W Cohen. 2020. Faithful embeddings for knowledge base queries. *Advances in Neural Information Processing Systems*, 33:22505–22516.

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.

Ricardo Usbeck, Axel-Cyrille Ngonga Ngomo, Bastian Haarmann, Anastasia Krithara, Michael Röder, and Giulio Napolitano. 2017. 7th open challenge on question answering over linked data (qald-7). In *Semantic Web Challenges: 4th SemWebEval Challenge at ESWC 2017, Portoroz, Slovenia, May 28-June 1, 2017, Revised Selected Papers*, pages 59–69. Springer.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Huandong Wang, Qiaohong Yu, Yu Liu, Depeng Jin, and Yong Li. 2021. Spatio-temporal urban knowledge graph enabled mobility prediction. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 5(4):1–24.

Tianyu Wu, Shizhu He, Jingping Liu, Siqi Sun, Kang Liu, Qing-Long Han, and Yang Tang. 2023. A brief overview of chatgpt: The history, status quo and potential future development. *IEEE/CAA Journal of Automatica Sinica*, 10(5):1122–1136.

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*.

Zhengcong Yin, Chong Zhang, Daniel W Goldberg, and Sathya Prasad. 2019. An nlp-based question answering framework for spatio-temporal analysis and visualization. In *Proceedings of the 2019 2nd International Conference on Geoinformatics and Data Analysis*, pages 61–65.

Xiaohan Zhang, Xinning Zhu, Jie Wu, Zheng Hu, and Chunhong Zhang. 2021. A framework of data fusion through spatio-temporal knowledge graph. In *International Conference on Knowledge Science, Engineering and Management*, pages 216–228. Springer.

A Appendix

A.1 Reproducibility Configuration

For STKG embedding, the dimension is set to D = 512. The model's parameters are updated using Adagrad (Duchi et al., 2011) with a learning rate of 0.1. The batch size is set to 1,000. STComplEx undergoes training for a maximum of 50 epochs, and the final parameters are determined based on the best validation performance.

During STKGQA, the parameters of the pre-trained language model and the STKG embeddings remain unchanged. The encoder is configured with 6 transformer layers, each consisting of 8 heads. The model's parameters are updated using Adam (Kingma and Ba, 2014) with a learning rate of 0.0002. The batch size is set to 150, and the number of epochs is 60. We employ the softmax function to transform the answer representation into probabilities. The model then updates its parameters by minimizing the cross-entropy loss function, aiming to assign a higher probability to the correct answer.

A.2 Baselines

A.2.1 STKG Embedding Baselines

ComplEx ComplEx (Trouillon et al., 2016) utilizes complex-valued embeddings and employs the Hermitian dot product, which is the complex counterpart of the standard dot product between real vectors. ComplEx does not incorporate spatio-temporal information during training and solely relies on basic triples.

TComplEx TComplEx (Lacroix et al., 2020) is a solution inspired by the canonical decomposition of tensors of order 4. It introduces novel regularization schemes and extends the capabilities of ComplEx. TComplEx is trained using facts that incorporate temporal information.

A.2.2 STKGQA Baselines

Pre-trained LMs BERT (Devlin et al., 2018) and RoBERTa (Liu et al., 2019) are pre-trained language models. We concatenate the question embedding and spatio-temporal information embedding, followed by learnable projections. The resulting embedding is scored using the dot product over all entities.

EmbedKGQA EmbedKGQA (Hu et al., 2017) is a framework for QA over regular KG. During training, we disregard question spatio-temporal annotations.

CronKGQA CronKGQA (Saxena et al., 2021) is a method based on TKGQA embedding. It first employs a language model model to obtain question embeddings and subsequently utilizes the scoring function of TKG embedding for answer prediction. In the experiment, we annotated time-related entities and utilized the model to predict answers.

TempoQR TempoQR (Mavromatis et al., 2022) utilizes a TKG embedding-based scoring function for answer prediction and incorporates additional temporal information. According to its method, we solely incorporate time information into the central entity, excluding spatial information. We find that incorporating spatial information diminishes the model's performance, given that it is trained based on TKG.

SubGTR Based on TempoQR, SubGTR (Chen et al., 2022b) extracts implicit information from temporal questions. It incorporates a semantic-aware and temporal inference module into the scoring function, so we also provide time clues and time constraints for the model.

A.3 Examples

Table 8 and Table 9 give two examples from the STQAD validation set. To facilitate potential follow-up work, we have annotated the entities within the model.

Table 8: The question contains three entities. The entity <*Stockholm*> implies location information, the entity <*Stockport_County_F.C.*> implies time information, and the entity <*George_Moncur*> serves as the central entity.

Question	Southwest of <Stockholm>, which team did <George_Moncur>play for posterior to the termination of <Stockport_County_F.C.>?
Paraphrased Question	After leaving Stockport County F.C., which team did George Moncur play for located southwest of Stockholm?
Answers	<Partick_Thistle_F.C.>
Entities	<Stockholm> <George_Moncur> <Stockport_County_F.C.>
Question Type	<playsFor>

Table 9: The question contains two entities. The entity <*Trinity_College_(Connecticut)*> contains time information, and the entity <*Algeria*> contains location information and is also a central entity.

Question	Name all countries created after <Trinity_College_(Connecticut)> and southeast of <Algeria>.
Paraphrased Question	Can you provide a list of countries that were established after Trinity College (Connecticut) and are located to the southeast of Algeria?
Answers	<Niger> <Libya>
Entities	<Trinity_College_(Connecticut)> <Algeria>
Question Type	<hasNeighbor>