Semantic Framework based Query Generation for Temporal Question Answering over Knowledge Graphs

Wentao Ding, Hao Chen, Huayu Li, Yuzhong Qu
State Key Laboratory for Novel Software Technology, Nanjing University, China
{wtding, haochen97}@smail.nju.edu.cn, lihuayuu@gmail.com, yzqu@nju.edu.cn

Abstract

Answering factual questions with temporal intent over knowledge graphs (temporal KGQA) attracts rising attention in recent years. In the generation of temporal queries, existing KGQA methods ignore the fact that some intrinsic connections between events can make them temporally related, which may limit their capability. We systematically analyze the possible interpretation of temporal constraints and conclude the interpretation structures as the Semantic Framework of Temporal Constraints, SF-TCons. Based on the semantic framework, we propose a temporal question answering method, SF-TQA, which generates query graphs by exploring the relevant facts of mentioned entities, where the exploring process is restricted by SF-TCons. Our evaluations show that SF-TQA significantly outperforms existing methods on two benchmarks over different knowledge graphs.

1 Introduction

With the rapid growth of knowledge graphs, temporal question answering over knowledge graphs (temporal KGQA) is attracting rising attention in recent years (Jia et al., 2018b, 2021). In temporal KGQA, a common phenomenon is that questions express temporal relations between events or time expressions, while knowledge graphs describe the facts resulting from each event. Existing methods handle the heterogeneity between natural language and knowledge graph representation in two ways. Some systems express temporal intents by constructing executable queries, some apply time-sensitive neural models to rank candidate answers. Considering that neural models are difficult to characterize the clear boundaries of concepts (e.g., exactly filter all events that occur "before 2022"), this paper focuses on generating queries that correspond to the meaning of questions.

From the logic perspective, formulated queries are actually logical restrictions about KG facts. The answers to a question is a set of KG objects, each of which satisfies the corresponding logical restrictions. In previous studies (e.g., Jia et al., 2018b), temporal intents are converted into restrictions over KG facts with quantitative time values. Example 1 illustrates a typical conversion from a temporal question to such restriction.

Example 1. *"Who was the president of the U.S. when John Lennon was shot?"*

The corresponding query on Wikidata can be formulated as the following logical restriction:

$$T_1 = \mathsf{time}(\mathsf{position\_held}(ANS, \mathsf{U.S.\_president}))$$
$$\wedge\, T_2 = \mathsf{time}(\mathsf{Murder\_of\_John\_Lennon})$$
$$\wedge\, \textsc{Overlaps}(T_1, T_2).$$

However, the idea of constructing queries with quantitative restrictions can not exhaust all possible scenarios. As illustrated in Example 2, facts with time values are not a necessary premise to introduce a temporal relation.

Example 2. *"Where was John Lennon standing when he was shot?"*

To construct a comparison restriction, we need to enumerate the "standing" of J.L. (i.e. all the experiences of his life). The enumeration is hard to implement and might introduce errors.[1] *In fact, the temporal intent does not rely on any time value. The two events occur simultaneously just because they are different aspects of the same entity (*wd:Q2341090*), the murder of John Lennon.*

The above example reveals that intrinsic connections can also make events temporally related. We argue that the neglect of such cases may limit the capability of existing methods. Therefore, the possible temporal constraints, especially those that do not rely on explicit time values, need to be specifically studied. The main challenges in concluding

[1]For example, Wikidata says that J.L.'s "residence"(wd:P551) includes Liverpool and New York, but does not provide the corresponding time duration.

Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 1867 - 1877
December 7-11, 2022

such constraints come from the complexity of natural language and the lack of supervision signals. Practical KGQA tasks often provide only question-answer pairs. i.e., the constraints on the relevant facts are unknown. Manually enumerating all possible constraint structures in a huge search space will be cumbersome or even infeasible. Thus, there is a need for a lightweight method to model the various constraints that correspond to possible temporal intents.

Inspired by the basic idea of frame semantics that "one cannot understand the meaning of a word without access to all the encyclopedic knowledge that relates to that word." (Fillmore et al., 2006), we assume that temporal intents are expressed as certain constraints about corresponding knowledge and could be interpreted by some structures over KG facts. Specifically, the events involved in a temporal constraint should provide certain KG facts, which support a possible interpretation of it. We conclude the temporal constraints and their corresponding interpretation structures as the *Semantic Framework of Temporal Constraints, SF-TCons*. SF-TCons describes what kinds of knowledge are needed and how they are composed in the potential interpretations. It consists of 6 interpretation structures, which will be presented in Section 2. To the best of our knowledge, SF-TCons is the first work to systematically summarize the interpretation structures for temporal KGQA tasks.

Based on SF-TCons, we propose a semantic-framework-based question answering method, SF-TQA, to convert SF-TCons into executable queries. SF-TQA generates query graphs by exploring the relevant facts of mentioned entities, where the query graph is a graph representation of executable logical queries that resembles subgraphs of KG (Yih et al., 2015). SF-TQA improves the accuracy of query generation by regarding SF-TCons as restrictions in the exploration. SF-TQA firstly evokes possible interpretations of temporal intents according to TimeML (Pustejovsky et al., 2010) annotations. It then grounds the temporal elements in corresponding interpretation structures by the relevant KG facts. The grounding phase will generate multiple candidate queries, the best candidate will be distinguished by ranking the pairs of questions and serialized queries with a BERT model.

The rest of this paper is organized as follows: Section 2 discusses the SF-TCons in detail. Section 3 presents SF-TQA. Section 4 evaluates the SF-TQA with two benchmarks over different knowledge graphs. Section 5 summarizes the related work. The last section concludes this paper.

2 Semantic Framework of Temporal Constraints

As previously introduced, temporal intents reflect constraints on events and time expressions. We argue that what really supports the constraints is the essential knowledge underlying the involved elements. For example, in a comparison like *"before WWI"*, what is needed is its start time *"1914"* rather than the named entity `wd:Q361` in KG. Therefore, temporal constraints can be interpreted by describing what kind of knowledge is needed and how they are composed. The interpretation structures of the constraints are presented as SF-TCons, the Semantic Framework of Temporal Constraints.

2.1 Temporal Constraints in Questions

Depending on whether the constraints concern quantitative attributes of a single event or the relations between events, we classify the temporal constraints as follows.

Value Constraints. The intentions about quantitative values are often expressed with time values or ordinals (e.g., "first president"). They require certain events to have corresponding temporal or ordinal attributes. Thus, they could be denoted as follow.

$$\textsc{HasValue}(E_1, T_1), \qquad \text{(VC-1)}$$
$$\textsc{HasValue}(E_1, O_1), \qquad \text{(VC-2)}$$

where E, T, O denotes events, time expressions and ordinals respectively. As an example, the intent "first president" could be denoted as $\textsc{HasValue}(\textit{"president"}, \textit{"first"})$. Specifically, temporal interrogatives (e.g., "when did sth. happen?") are denoted as $\textsc{HasValue}(E_1, T?)$, which declare the existence of the temporal attributes but has no restrict on the specific value.

Relation Constraints. The possible relations between time and events have been well studied in the AI area. We follow TimeML, the most commonly used annotation specification, to model the relation constraints.

Example 3. *TimeML-style annotations for the question in Example 2 :*
Where was John Lennon [$_{\text{Event}_1}$ standing] [$_{\text{Signal}_1}$

when] he was [$_{\text{Event}_2}$ shot]?
⟨TLINK reltype=SIMULTANEOUS target=EVENT$_1$ relatedTo=EVENT$_2$ signal=SIGNAL$_1$ /⟩

As illustrated in Example 3, temporal relations are triggered by certain signals (e.g., "when") and classified into pre-defined reltypes. For the practical demand of QA tasks, we formalized the relation constraints as

$$\text{RELATION}(\mathcal{T}_R, E_1, T_1), \quad \text{(RC-1)}$$
$$\text{RELATION}(\mathcal{T}_R, E_1, E_2), \quad \text{(RC-2)}$$

where $\mathcal{T}_R$ denotes the 13 temporal *reltype*s in TimeML (Pustejovsky et al., 2003), E and T denotes events and time expressions respectively. The TimeML-style annotation in the example question corresponds the following RC-2 constraint:

$$\text{RELATION}(\text{SIMULTANEOUS}, \textit{"standing"}, \textit{"shot"})$$

2.2 Interpretation Structure for Temporal Constraints

As previously mentioned, one temporal constraint could be supported by various interpretations. We summarize 6 interpretation structures (IS) according to whether the involved event expressions are intrinsically connected and what connector between them can correspond to the expected meanings. In order to enhance the generality of the IS as much as possible, we do not restrict the specific semantic representations of involved events, but only focus on the key knowledge that they can provide. The 6 IS are presented as follows.

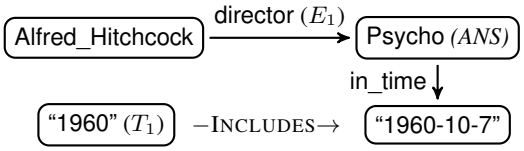
IS-1 Comparison structure

$$\text{HASVALUE}(E_1, T_1) \,|\, \text{RELATION}(\mathcal{T}_r, E_1, T_1|E_2) \Rightarrow \text{COMPARE}\langle \circ, \text{time}(E_1), \text{time}(T_1|E_2)\rangle$$

This structure interprets **VC-1** and **RC**, where $\circ$ denotes algebraic predicate for time values (Allen, 1983; Jia et al., 2018b). Specifically, the predicate $\circ$ is required to be EQUAL in **VC-1** and is determined according to the identified type $\mathcal{T}_r$ in **RC**. This structure supposes that the involved events provides certain time values.

For example, the question: "*Which movie did Alfred Hitchcock* [$_{\text{Event}_1}$ *direct*] [$_{\text{Signal}_1}$ *in*] [$_{\text{Time}_1}$ *1960*]*?*" corresponds to the following constraint and KG facts, where the "direct" event provides the value "1960-10-7".

$$\text{COMPARE}\langle \text{INCLUDES}, \text{time}(\textit{"direct"}), \textit{"1960"}\rangle$$

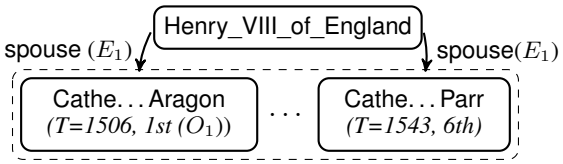


IS-2 Ordering Structure

$$\text{HASVALUE}(E_1, O_1) \Rightarrow \text{ORDER}\langle \text{attr}(E_1), O_1\rangle$$

This structure interprets **VC-2** by ordering entities (or facts) that are described by E_1. It supposes that E_1 describes a common attribute of certain objects to be ordered. For example, the question: *"When did Henry the VIII* [$_{\text{Event}_1}$ *marry*] *his* [$_{\text{Ordinal}_1}$ *first*] *wife?"* corresponds to

$$\text{ORDER}\langle \text{attr}(\textit{"marry"}), \textit{"first"}\rangle$$



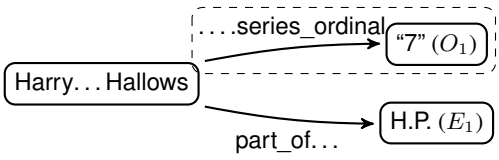
IS-3 Direct Query Structure

$$\text{HASVALUE}(E_1, X) \Rightarrow \text{FIND}\langle \text{ent}(E_1), \text{attr}(X)\rangle$$

In some cases, the expected values are directly represented in KG facts. This structure interprets **VC** by directly finding the expected value X in certain attributes of some related entity. It supposes that the entity is related to the mentioned event E_1.

For example, the description: *"... did the* [$_{\text{Ordinal}_1}$ *7th*] [$_{\text{Event}_1}$ *Harry Potter book*] ... " corresponds to the following representation and KG facts, where the entity "Harry Potter and the Deathly Hallows" has some attribute with the value "7".

$$\text{FIND}\langle \text{ent}(\textit{"... book"}), \text{attr}(\textit{"7th"})\rangle$$



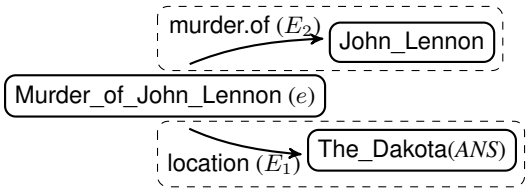
IS-4 Same Entity Structure

$$\text{RELATION}(\mathcal{T}_r, E_1, E_2) \Rightarrow \text{SAMEENTITY}\langle e, \text{attr}(E_1), \text{attr}(E_2)\rangle$$

This structure interprets *simultaneous* cases of **RC-2**. It supposes that the events should be attributes of a certain entity e.

For example, the previously introduced question *"Where was John Lennon* [$_{\text{Event}_1}$ *standing*] [$_{\text{Signal}_1}$ *when*] *he was* [$_{\text{Event}_2}$ *shot*]*?"* corresponds to

$$\textsc{SameEntity}\langle e, \text{attr}(\textit{"standing"}), \text{attr}(\textit{"shot"})\rangle$$



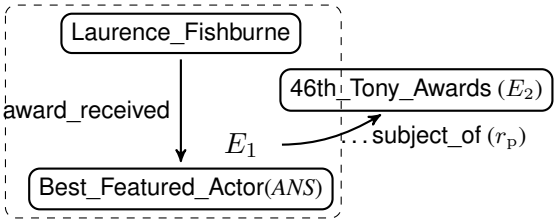
IS-5 Part-of Structure

$$\textsc{Relation}(\mathcal{T}_r, E_1, E_2) \Rightarrow \textsc{PartOf}\langle r_{\text{p}}, E_1, E_2\rangle$$

This structure interprets *including* cases of **RC-2**. It does not restrict the representation of events E_1 and E_2 in KG, but requires that their representation must be connected by a relation r_{p} which implies "part-of".

For example, the question *"What award did Laurence Fishburne* [$_{\text{Event}_1}$ *received*] [$_{\text{Signal}_1}$ *at*] [$_{\text{Event}_2}$ *the 46th Tony Awards*]*?"* corresponds to the following representation and KG facts, where E_1 corresponds to a *statement*[2] and E_2 corresponds to a named entity.

$$\textsc{PartOf}\langle r_{\text{p}}, \textit{"received"}, \textit{"...Awards"}\rangle$$



IS-6 Sequent Structure

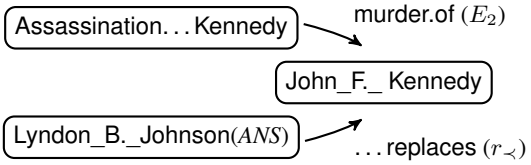
$$\textsc{Relation}(\mathcal{T}_r, E_1, E_2) \Rightarrow \textsc{Sequent}\langle r_{\prec}, \text{ent}(E_1), \text{ent}(E_2)\rangle$$

This structure interprets *before/after* cases of **RC-2**. It supposes the events make a pair of related entities to be sequential in KG, where the entities are involved in E_1 and E_2 respectively and they must be connected by a relation $r_{\prec}$ which indicates a preceding (or succeeding) relation.

[2]Statement is a Wikidata format for representing complex items. It can be roughly considered as RDF blank node.

For example, the question *"Who* [$_{\text{Event}_1}$ *became*] *the president* [$_{\text{Signal}_1}$ *after*] *J.F. Kennedy was* [$_{\text{Event}_2}$ *shot*]*?"* corresponds to

$$\textsc{Sequent}\langle r_{\prec}, \text{ent}(\textit{"became"}), \text{ent}(\textit{"shot"})\rangle$$



In summary, IS 1 to 3 interpret the temporal constraints via temporal facts with explicit quantitative values. IS 4 to 6 model the intrinsic connections that can make events temporally related. It is worth noting that SF-TCons only expresses the expected form of corresponding knowledge, how to obtain the specific knowledge is left to the implementation of question-answering systems.

3 Semantic-Framework-Based Temporal Question Answering

Figure 1 illustrates the question-answering process of the semantic-framework-Based temporal question answering method, SF-TQA. The query generation consists of two steps, 1) evoking the constraints and their possible interpretations (i.e., *constraint evocation*) and 2) grounding the constraints by exploring the relevant KG facts (i.e., *constraint grounding*). The generated candidate queries will be ranked by a BERT model, and the execution results of the highest-scored query will be considered as answers.

3.1 Constraint Evocation

The first step of SF-TQA is to determine the possible constraints. We fine-tune a BERT model to annotate the temporal elements. The corresponding constraints and interpretation structures are evoked according to recognized signals. The elements that involve certain constraints are determined by TimeML relations or by simply taking the temporal elements that are directly described by the signals (i.e., the nearest neighbor of the corresponding signals). The algebraic predicates in the comparison structure are determined by normalizing the TimeML relation types, while other implicit elements are left to the grounding phase.

3.2 Constraint Grounding

In general query graph generation, basic query graphs are constructed as 1 or 2 hop paths from

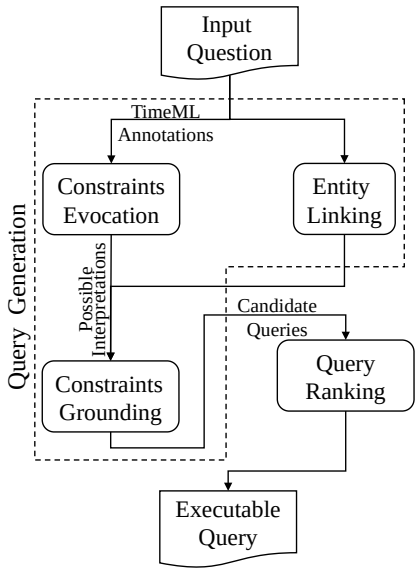


Figure 1: The question-answering process of SF-TQA.

mentioned entities to answers, and they are extended by pre-defined expanding action (Yih et al., 2015) or fixed interpretation structures of constraints (Bao et al., 2016). In temporal KGQA, the main issue is that events could have various representations in KG. As illustrated by the examples in Section 2.2, they could be represented by named entities, triplet facts, or attributes of their participants. Therefore, we treat the generation of query graphs as grounding the temporal elements in the interpretations of SF-TCons. We divide the descriptions of events into *nominal* and *predicative*. We suppose that nominal descriptions could be the event themselves, and predicative descriptions reflect certain aspects of the events, such as their participants or their post-effect. Therefore, nominal events could be linked entities and others correspond to the neighboring nodes or facts of the explored subgraph(s) or linked entities. The corresponding nodes or facts must provide the knowledge required by corresponding interpolation structures.

We illustrate the above process by the example in Figure 2. In this example, the entity linking module will provide John_Lennon as a linked entity, and the grounding start with the "shot" event which contains the only linked entity. We will explore *all* the neighboring facts of John_Lennon (as illustrated in Figure 2) as candidates for the event. Since "shot" is a predicative event and the SAME_ENTITY constraint requires it to provide an attribute, we will find a triplet that contains John_Lennon and take the other entity in it (i.e., Murder_of_John_Lennon) as the expected e. Similarly, we explore the neighboring facts and select one relation that matches with the question meaning (i.e., location for "standing"). When there are multiple candidate relations, we will rank the candidates by scoring their serializations with a BERT model. The highest-scored one will be filled in the corresponding slot.

In the specific implementation, which candidates satisfy the question meanings best are determined by neural models. In the training process, we take relations that appear on shortest paths between mentioned entities and answers as positive samples. In particular, the relation that entails *part of* or *precedes* are filtered according to the KG schema in the training process and are predicted by neural models during the test process. Queries for the questions of multiple constraints are the conjunction of the grounding result of each constraint and queries for the questions with no temporal constraints are unrestricted basic query graphs.

3.3 Query Ranking

SF-TQA usually generates multiple candidate queries for one question. We select one of the candidates via neural ranking models. Specifically, we express the generated queries via SPARQL [3] and serialize the queries by dropping auxiliary symbols (e.g., "{"). We use the BERT model with cross-entropy loss to score the pair of the input question and serialized queries. For each question, we use the candidate queries with the highest F_1 score as the positive samples and select k others as negative samples. In order to make our model more robust, we classify the negatives samples as *confusing queries* and *irrelevant queries*. Confusing queries are those that can find partial answers but of lower F_1 scores than the positive samples. Irrelevant queries are those whose outputs have no intersection with the correct answers. The ratio of confusing queries to irrelevant queries is $1:1$. The necessity of classifying the negative sample is presented in Appendix A.

4 Evaluation

4.1 Datasets

We evaluate our method on **TempQuestions** (Jia et al., 2018a) and **TimeQuestions** (Jia et al., 2021) with the 2015-08-09 dumps of Freebase[4] and the

[3] https://www.w3.org/TR/sparql11-query/
[4] https://developers.google.com/freebase

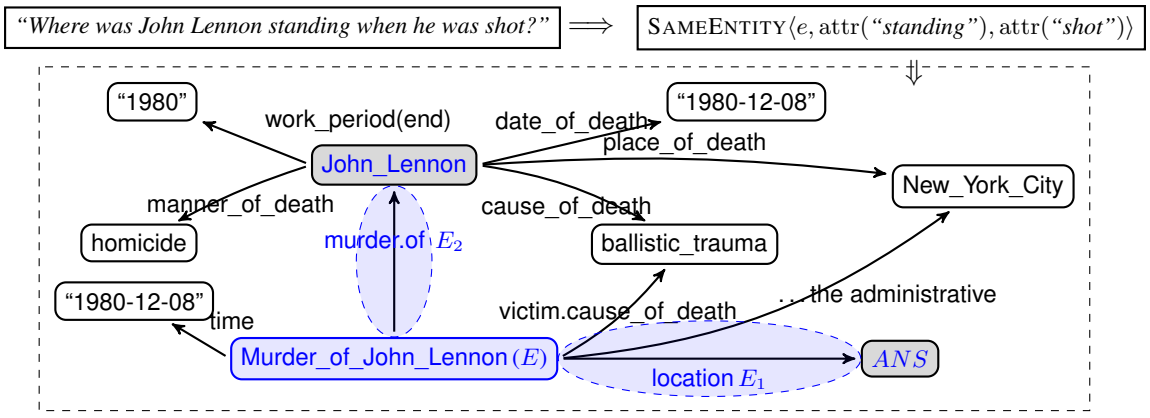


Figure 2: The possible generation process for question in Example 2. The linked entity and expected answer are colored in gray and the facts that can satisfies the evoked interpretation are highlighted in blue.

2019-01-28 dumps of Wikidata[5], respectively. The statistics of the benchmarks are described in Table 1. Since TempQuestions does not give the partition between training data and test data, we randomly divided it with the same ratio as TimeQuestions.

Dataset	Train	Dev.	Test
TempQuestions	762	254	255
TimeQuestions	9,708	3,236	3,237

Table 1: The statistics of questions in benchmarks.

4.2 Evaluation Metrics

We report the Hit@1 (denoted as $H@1$), Precision (denoted as Pr) , Recall (denoted as Re) and F_1 score of the evaluation results. Our computation follows Jia et al.'s (2021)[6], where the precision is considered 1 if the output of a question is empty and the F_1 score on a dataset is computed as the average of the scores of each question.

4.3 Compared Methods

On TempQuestions, we compare our results with general KGQA methods AQQU (Bast and Haussmann, 2015), QUINT (Abujabal et al., 2017) and their improved version(Jia et al., 2018b) by incorporating the temporal question decomposition method TEQUILA, QUINT+TEQUILA and AQQU+TEQUILA. On TimeQuestions, we compare our results with general KGQA methods PullNet (Sun et al., 2019), GraftNet (Sun et al., 2018), UNIQORN (Pramanik et al., 2021) and the temporal KGQA method EXAQT (Jia et al., 2021).

4.4 Implementation Details

Our results are obtained on a workstation with an Intel Xeon Gold 5222 CPU, 32 GB of RAM, and NVIDIA RTX3090 GPUs. The hyper-parameters of the ranking models are listed in Table 2. They are determined according to the F_1 scores on the development sets. We use ELQ (Li et al., 2020) for entity linking. We randomly sample 5,000 questions from the training set of TimeQuestions to fine-tune a BERT model for TimeML annotations. The questions are firstly automatically annotated by simple regex according to their POS-tags and surface forms (e.g., verb tokens may indicate event) then corrected by human annotators. The types of TimeML relations are determined by normalizing the signals via manual rules (e.g., "during" corresponds to INCLUDES).

	TempQ.	TimeQ.
Learning Rate	$5e-5$	$5e-5$
Batch size	8	8
Epochs	10	15
Pos./Neg. Ratio	1 : 20	1 : 25

Table 2: Hyper parameters for the ranking model.

4.5 Main Results

Table 3 and 4 report the results of compared methods on TempQuestions and TimeQuestions respectively.[7] Our method, SF-TQA, achieves the best

[5] https://archive.org/download/wikibase-wikidatawiki-20190128

[6] Their script could be downloaded from here.

[7] The execution results of compared methods on TempQuestions are obtained from Jia et al.'s (2018b) homepage

Method	H@1	Pr	Re	F_1
QUINT	27.0	30.3	**51.2**	28.8
+TEQUILA	31.7	40.7	42.2	32.0
AQQU	24.9	26.6	48.8	27.2
+TEQUILA	36.2	40.3	43.4	37.5
SF-TQA	**41.2**	**42.2**	48.8	**41.1**

Table 3: Results (%) on **TempQuestions**. The best results are in **bold** and the second bests are underlined.

Method	H@1	Pr	Re	F_1
PullNet	10.5	5.4	19.2	7.6
GraftNet	45.2	52.7	45.2	37.8
UNIQORN	33.1	14.8	45.4	19.5
EXAQT	**56.5**	**59.3**	56.9	45.6
SF-TQA	53.9	55.1	**62.1**	**52.7**

Table 4: Results (%) on **TimeQuestions**. The best results are in **bold** and the second bests are underlined.

results on both two benchmarks. Specially, we improve the F_1 scores by $+3.6$ and $+7.1$ points on TempQuestions and TimeQuestions respectively. On TempQuestions, SF-TQA improves the Hit@1 and precision by $+5.0$ and $+1.9$ respectively. On TimeQuestions, SF-TQA achieves better recall (5.2 points higher) while EXAQT achieves better precision (4.2 points higher). The reason could be the different strategies when dealing with unsolvable problems. EXAQT tends to output empty answers (which correspond to 1 in precision) and SF-TQA degrades to the unrestricted generation of basic query graphs (which capture incomplete question meanings). The Hit@1 of SF-TQA is 2.6 points lower than EXAQT might because EXAQT ranks all candidate answers while SF-TQA just randomly returns one candidate that satisfies the generated query.

4.6 Ablation Studies

We conduct ablations on the necessity of interpretations for intrinsic connections (i.e., IS-4 to 6). We analyze the result on questions with only relation constraints. The ablation results are illustrated in Table 5. The 2nd row shows that without IS 4 to 6 the F_1 scores drop 1.5 and 3.8 points respectively on the benchmarks. The 3rd row shows that results obtained by generating only basic query graphs without any restriction will decrease the F_1 scores by 14.1 and 4.7 points respectively. The differences between the results on the two benchmarks might reflect the differences between underlying KGs. SF-TQA without IS 4 to 6 achieves acceptable results on TempQuestions, which might indicate that Freebase can provide sufficient temporal facts for comparisons. SF-TQA with only basic query graphs on TimeQuestions performs much better than on TempQuestions, which might indicate that Wikidata provides richer and finer relations between entities, thus the connections between mentioned entities and answers are more likely to be satisfied via simple relation paths.

Method	TempQ.		TimeQ.	
	H@1	F_1	H@1	F_1
Full System	**37.5**	**38.1**	**41.3**	**40.7**
w/o IS-4 to 6	**37.5**	36.6	35.4	36.9
w/o IS	29.2	24.0	34.7	36.0

Table 5: Results (%) for the effectiveness of interpretation structures on relation constraints.

4.7 Error Analysis

We analyze the main errors of 100 questions of which the F_1 scores are less than 1. The results are illustrated in Table 6.

Main Error	TempQ.	TimeQ.
Recognition Errors	12%	36%
Uncovered Constraints	14%	20%
Ranking Errors	26%	2%
Inconsistent Answers	10%	8%
Incomplete Knowledge	38%	34%

Table 6: The statistics of main errors of sampled questions.

The 1st row counts the questions with incorrectly recognized entities or temporal constraints, which reveals that SF-TQA severely suffers from error propagations on TimeQuestions. The 2nd row counts the questions whose meaning can not be perfectly expressed by generated constraints (e.g., questions with multi-hop non-temporal property paths like *"wife of the actor who played in the movie pinball wizard"*). The 3rd row shows that our ranking model is hard to train with limited data (TempQuestions contains less

and the reported results of compared methods on TimeQuestions are provided by the authors of EXAQT (Jia et al., 2021).

than 1,000 training samples). Besides, data quality appears to be an important issue. In about 10% of the sample questions, the provided answers are inconsistent with the knowledge in the given KG. For example, TimeQuestions annotate 2010_F1_Championship (wd:Q69934) as the answer of *"Who won the 2010 f1 championship?"*. For over 1/3 of the sampled questions, KG can not provide sufficient evidence (e.g., occurrence times of the corresponding facts are not provided) for obtaining all answers.

5 Related work

Temporal Information in Natural Language. Temporal information has attracted the attention of AI and linguistics communities for a long time. Allen presents an interval-based temporal logic for reasoning the relation between time duration (Allen, 1983) and a computation theory for action and time (Allen, 1984). He concludes 13 possible interval relations with their transitivity table. Mani and Wilson (2000) introduces an annotation scheme for temporal expression in news and discusses its possible application in event ordering and event time alignment. TimeML (Pustejovsky et al., 2003, 2010) is a specification for annotating temporal information from narratives. TimeML has become the de facto standard in the NLP community. It annotates time expressions, events, the relations between them, and the signals that trigger the relations in XML form.

Temporal KGQA. Early KGQA systems usually do not handle temporal constraints (e.g., Berant and Liang, 2014) or apply simple heuristics about their surface forms (Berant et al., 2013; Bast and Haussmann, 2015; Bao et al., 2016). Some benchmarks that specifically focus on temporal intents in KGQA emerge in recent years, including TempQuestions (Jia et al., 2018b), TimeQuestions (Jia et al., 2021) and TempQA-WD (Neelam et al., 2022). In terms of the technologies for temporal KGQA, Jia et al. (2018a) proposes TEQUILA. It relies on limited hand-crafted rules to decompose complex temporal relations and solves composed simple questions via underlying general KGQA systems. EXAQT (Jia et al., 2021) uses Group Steiner Trees to anchor a KG sub-graph for each question, retrieving answers in the sub-graph with augmented temporal facts by an RGCN model. Besides, there are also some researches specifically focus on question event-centric or temporal knowledge graphs. Costa et al. (2020) proposes a question answering benchmark Event-QA over EventKG (Gottschalk and Demidova, 2018, 2019). Saxena et al. (2021) proposes CronQuestion over a sub-graph of Wikidata with a limited subset of relations for evaluating temporal KG embeddings.

In summary, existing temporal KGQA methods either analyze only the surface form of temporal constraints or rely on end-to-end neural models. While neural models might be robust to diversified representations of knowledge, they are hard to characterize the clear boundaries of temporal constraints (e.g., accurately filtering all events that occur before 2022).

KGQA via Query Graph Generation. Constructing queries via exploring the relevant facts of mentioned entities is a common practice in KGQA, especially in the situations where only question-answers pairs are provided. Yih et al. (2015) defines query graphs that can be straightforwardly mapped to an executable logical query. They model the generation of query graphs as a staged search problem, where the query graphs are expanded by exploring legitimate predicate sequences starting from mentioned entities. Bao et al. (2016) expands basic query graphs with 6 kinds of manually designed constraints including quantitative temporal and ordinal constraints. Luo et al. (2018) encodes query graphs of complex structures into a uniform vector representation for complex questions. Lan and Jiang (2020) prunes the search space via early incorporation of constraints.

The existing query graph generation methods are not specifically designed for temporal constraints, they simply suppose that temporal or ordinal signals correspond to quantitative constraints. Specifically, Bao et al. (2016), Luo et al. (2018), and Lan and Jiang (2020) recognize time constraints via syntax signals and simply interpret them as general aggregation functions (e.g., greater than X, max at N), i.e., their interpretations of temporal constraints are similar to "SF-TQA w/o IS-4 to 6" (referring to Table 5). In contrast, we systematically analyze the interpretation structure of temporal constraints, including the analysis of what kind of intrinsic connection can make events temporally related.

6 Conclusion and Future work

In this paper, we study the logical constraint that corresponds to temporal intents in questions. Our main contributions can be summarized as follows:

- We propose the idea of analyzing temporal intents via possible interpretation structures. We conclude the interpretation structures as SF-TCons, which allows one constraint expression has various interpretations.
- We propose the semantic-framework-based temporal question answering method, SF-TQA. SF-TQA mitigates the heterogeneity between expressions of temporal intents and KG facts. It enhances the query generation via structural restrictions provided by SF-TCons.
- Our implementation of SF-TQA establishes new SOTAs on two benchmarks and improves the F_1 scores by $+3.6$ and $+7.1$ points respectively.

In the near future, it is worth exploring to alleviate the possible knowledge incompleteness in practical KG by developing a hybrid question-answering method on both knowledge graphs and web texts. In addition, this paper focuses only on temporal intent, while problems in real configurations may contain both complex non-temporal and temporal intents. Therefore, it would be helpful to combine SF-TCons with general KGQA systems for complex questions.

Limitations

- Due to the compositionality of natural language, a temporal question could be very complex, which is beyond the ability of our implemented QA system. For example, The following question is syntactically legitimate but can not be handled by SF-TQA: *"What year did the second president of the United States, elected after the last spouse of the author of 'Wish Tree for Washington, DC' was shot, marry his wife?"*
- While the linguistic and entity annotations help SF-TQA alleviate the lack of structured supervision, they make it hard to apply SF-TQA to low-resource languages or questions with no named entities (e.g., *"What are the important events that will happen at the turn of the century?"*). Besides, as a pipeline method, SF-TQA suffers from possible error propagations.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (NSFC) under Grant No. 62072224. and the Program B for Outstanding PhD candidate of Nanjing University. The authors would like to thank all the participants of this work and anonymous reviewers.

References

Abdalghani Abujabal, Rishiraj Saha Roy, Mohamed Yahya, and Gerhard Weikum. 2017. QUINT: interpretable question answering over knowledge bases. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017 - System Demonstrations*, pages 61–66. Association for Computational Linguistics.

James F. Allen. 1983. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11):832–843.

James F. Allen. 1984. Towards a general theory of action and time. *Artificial intelligence*, 23(2):123–154.

Junwei Bao, Nan Duan, Zhao Yan, Ming Zhou, and Tiejun Zhao. 2016. Constraint-based question answering with knowledge graph. In *Proceedings of 26th International Conference on Computational Linguistics: Technical Papers, COLING 2016, Osaka, Japan, December 11-16, 2016*, pages 2503–2514. ACL.

Hannah Bast and Elmar Haussmann. 2015. More accurate question answering on freebase. In *Proceedings of the 24th ACM International Conference on Information and Knowledge Management, CIKM 2015, Melbourne, VIC, Australia, October 19 - 23, 2015*, pages 1431–1440. ACM.

Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. 2013. Semantic parsing on freebase from question-answer pairs. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1533–1544. ACL.

Jonathan Berant and Percy Liang. 2014. Semantic parsing via paraphrasing. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, Volume 1: Long Papers*, pages 1415–1425. The Association for Computer Linguistics.

Tarcísio Souza Costa, Simon Gottschalk, and Elena Demidova. 2020. Event-qa: A dataset for event-centric question answering over knowledge graphs. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management, CIKM 2020, Virtual Event, Ireland, October 19-23, 2020*, pages 3157–3164. ACM.

Charles J Fillmore et al. 2006. Frame semantics. *Cognitive linguistics: Basic readings*, 34:373–400.

Simon Gottschalk and Elena Demidova. 2018. Eventkg: A multilingual event-centric temporal knowledge graph. In *Proceedings of the 15th Extended Semantic Web Conference, ESWC 2018, Heraklion, Crete, Greece, June 3-7, 2018*, volume 10843 of *Lecture Notes in Computer Science*, pages 272–287. Springer.

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

Zhen Jia, Abdalghani Abujabal, Rishiraj Saha Roy, Jannik Strötgen, and Gerhard Weikum. 2018a. Tempquestions: A benchmark for temporal question answering. In *Companion Proceedings of the The Web Conference 2018, WWW '18, Lyon , France, April 23-27, 2018*, pages 1057–1062. ACM.

Zhen Jia, Abdalghani Abujabal, Rishiraj Saha Roy, Jannik Strötgen, and Gerhard Weikum. 2018b. TEQUILA: temporal question answering over knowledge bases. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management, CIKM 2018, Torino, Italy, October 22-26, 2018*, pages 1807–1810. ACM.

Zhen Jia, Soumajit Pramanik, Rishiraj Saha Roy, and Gerhard Weikum. 2021. Complex temporal question answering on knowledge graphs. In *Proceedings of the 30th ACM International Conference on Information and Knowledge Management, CIKM 2021, Virtual Event, Queensland, Australia, November 1 - 5, 2021*, pages 792–802. ACM.

Yunshi Lan and Jing Jiang. 2020. Query graph generation for answering multi-hop complex questions from knowledge bases. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 969–974. Association for Computational Linguistics.

Belinda Z. Li, Sewon Min, Srinivasan Iyer, Yashar Mehdad, and Wen-tau Yih. 2020. Efficient one-pass end-to-end entity linking for questions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 6433–6441. Association for Computational Linguistics.

Kangqi Luo, Fengli Lin, Xusheng Luo, and Kenny Q. Zhu. 2018. Knowledge base question answering via encoding of complex query graphs. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2185–2194. Association for Computational Linguistics.

Inderjeet Mani and D. George Wilson. 2000. Robust temporal processing of news. In *Proceedings of 38th Annual Meeting of the Association for Computational Linguistics, Hong Kong, China, October 1-8, 2000*, pages 69–76. ACL.

Sumit Neelam, Udit Sharma, Hima Karanam, Shajith Ikbal, Pavan Kapanipathi, Ibrahim Abdelaziz, Nandana Mihindukulasooriya, Young-Suk Lee, Santosh K. Srivastava, Cezar Pendus, Saswati Dana, Dinesh Garg, Achille Fokoue, G. P. Shrivatsa Bhargav, Dinesh Khandelwal, Srinivas Ravishankar, Sairam Gurajada, Maria Chang, Rosario Uceda-Sosa, Salim Roukos, Alexander G. Gray, Guilherme Lima, Ryan Riegel, Francois P. S. Luus, and L. Venkata Subramaniam. 2022. A benchmark for generalizable and interpretable temporal question answering over knowledge bases. *CoRR*, abs/2201.05793.

Soumajit Pramanik, Jesujoba Alabi, Rishiraj Saha Roy, and Gerhard Weikum. 2021. UNIQORN: unified question answering over RDF knowledge graphs and natural language text. *CoRR*, abs/2108.08614.

James Pustejovsky, José M. Castaño, Robert Ingria, Roser Saurí, Robert J. Gaizauskas, Andrea Setzer, Graham Katz, and Dragomir R. Radev. 2003. Timeml: Robust specification of event and temporal expressions in text. In *New Directions in Question Answering*, pages 28–34. AAAI Press.

James Pustejovsky, Kiyong Lee, Harry Bunt, and Laurent Romary. 2010. Iso-timeml: An international standard for semantic annotation. In *Proceedings of the International Conference on Language Resources and Evaluation, LREC 2010, 17-23 May 2010, Valletta, Malta*. European Language Resources Association.

Apoorv Saxena, Soumen Chakrabarti, and Partha P. Talukdar. 2021. Question answering over temporal knowledge graphs. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL-IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 6663–6676. Association for Computational Linguistics.

Haitian Sun, Tania Bedrax-Weiss, and William W. Cohen. 2019. Pullnet: Open domain question answering with iterative retrieval on knowledge bases and text. 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 2019, Hong Kong, China, November 3-7, 2019*, pages 2380–2390. Association for Computational Linguistics.

Haitian Sun, Bhuwan Dhingra, Manzil Zaheer, Kathryn Mazaitis, Ruslan Salakhutdinov, and William W. Cohen. 2018. Open domain question answering using early fusion of knowledge bases and text. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018, Brussels, Belgium, October 31 - November 4, 2018*, pages 4231–4242. Association for Computational Linguistics.

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 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers*, pages 1321–1331. Association for Computer Linguistics.

A Appendix for Different Training Strategies

We also evaluated the effects of different training strategies as illustrated in Table 7. The results in 2 to 4 rows are obtained by simply using the irrelevant negatives, confusing negatives or randomly sampling negatives without classification respectively. The results show that both of the two types of negative sample are needed for training. The balanced sampling of the two types effectively improves SF-TQA on the smaller dataset, TempQuestions.

Method	TempQ.		TimeQ.	
	H@1	F_1	H@1	F_1
Full System	**41.2**	**41.1**	**53.9**	**52.7**
w/o confusing	34.9	35.9	49.5	49.3
w/o irrelevant	10.6	10.4	37.5	36.0
random neg.	37.3	37.4	53.5	52.6

Table 7: Results (%) of different sampling strategies for the negative samples.