TIQ: A Benchmark for Temporal Question Answering with Implicit Time Constraints

Zhen Jia
Southwest Jiaotong University
Chengdu, China
zjia@swjtu.edu.cn

Philipp Christmann
Max Planck Institute for Informatics
Saarland Informatics Campus
Saarbruecken, Germany
pchristm@mpi-inf.mpg.de

Gerhard Weikum
Max Planck Institute for Informatics
Saarland Informatics Campus
Saarbruecken, Germany
weikum@mpi-inf.mpg.de

ABSTRACT

Temporal question answering (QA) involves explicit (e.g., *"... before 2024"*) or implicit (e.g., *"... during the Cold War period"*) time constraints. Implicit constraints are more challenging; yet benchmarks for temporal QA largely disregard such questions. This shortcoming spans three aspects. First, implicit questions are scarce in existing benchmarks. Second, questions are created based on hand-crafted rules, thus lacking diversity in formulations. Third, the source for answering is either a KB or a text corpus, disregarding cues from multiple sources. We propose a benchmark, called TIQ (**T**emporal **I**mplicit **Q**uestions), based on novel techniques for constructing questions with implicit time constraints. First, questions are created automatically, with systematic control of topical diversity, timeframe, head vs. tail entities, etc. Second, questions are formulated using diverse snippets and further paraphrasing by a large language model. Third, snippets for answering come from a variety of sources including KB, text, and infoboxes. The TIQ benchmark contains 10,000 questions with ground-truth answers and underlying snippets as supporting evidence.

CCS CONCEPTS

• **Information systems** → *Question answering*.

KEYWORDS

Question Answering, Temporal Questions, Benchmarks

ACM Reference Format:
Zhen Jia, Philipp Christmann, and Gerhard Weikum. 2024. TIQ: A Benchmark for Temporal Question Answering with Implicit Time Constraints . In *Companion Proceedings of the ACM Web Conference 2024 (WWW '24 Companion), May 13–17, 2024, Singapore, Singapore.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3589335.3651895

1 INTRODUCTION

Motivation. Question answering (QA) aims to obtain crisp answers to natural language questions posed by end users [17]. One special case of QA is temporal QA, which focuses on questions with temporal constraints and has recently found increasing interest [8, 10, 18]. Temporal constraints consist of a temporal expression [19] and a temporal relation (like before, after, or during) indicating the temporal intent of user information needs. The temporal expression can be a specific date (e.g., *"February 27, 2024"*) or implicitly represented by an event (e.g., *"WWW 2024"*), or a phrase (e.g., *"introduction of the Euro"*). To provide accurate answers QA systems need to identify such explicit or implicit temporal constraints and deduce the relation between the temporal scopes of candidate answers and the constraints. Questions with implicit constraints, or *implicit questions*, are most challenging with the difficulty of understanding and resolving the time scope of the constraints. Consider the following implicit question as an example:

q_1: *Which football club did Messi join after Paris Saint-Germain?*

The corresponding *explicit question* would be

q_2: *Which football club did Messi join in July 2023?*

While there exists a wide range of benchmarks for temporal QA [1, 4–6, 8, 9, 11, 18, 20, 22], the existing datasets are limited in evaluating the capabilities of QA systems in answering implicit questions: (i) implicit questions are typically scarce which means that performance deficits are neglectable, (ii) questions are constructed based on few handcrafted templates, and (iii) questions are derived from a specific knowledge source in mind.

Table 1: Comparison of benchmarks for temporal QA.

Benchmark	No. of implicit questions	Knowledge source		
		KB	Text	Infobox
TIME-SENSITIVE QA [1]	–	✓	✓	✗
STREAMINGQA [9]	–	✗	✓	✗
TEMPQUESTIONS [6]	209	✓	✗	✗
TIMEQUESTIONS [8]	1,476	✓	✗	✗
TEMPQA-WD [11]	154	✓	✗	✗
TEMPTABQA [5]	7,242	✗	✗	✓
CRONQUESTIONS [18]	91,165 (5 KB-relations)	✓	✗	✗
TEMPREASON [20]	21,877 (10 KB-relations)	✓	✗	✗
TIQ (ours)	10,000	✓	✓	✓

A new benchmark. To overcome these limitations, we propose a method for automated construction of implicit questions. To ensure that questions are not specific to a single input source, our method taps into multiple sources: Wikipedia text and infoboxes, and the Wikidata KB. The method is configurable to control different aspects including (i) the temporal scope of questions, (ii) topical domains (sports, politics, etc.), (iii) diversity of topic entities, (iv) ratio of prominent vs. long-tail entities, (v) question complexity (number of entities and token length in questions), (vi) total number of questions, and other factors.

Our method for benchmark construction has four stages:

WWW '24 Companion, May 13–17, 2024, Singapore, Singapore

ACM ISBN 979-8-4007-0172-6/24/05.
https://doi.org/10.1145/3589335.3651895

(i) **Topic entity sampling:** select topic entities from Wikipedia year pages;
(ii) **Information snippet retrieval:** retrieve temporal snippets for each topic entity from Wikipedia text, Wikipedia infoboxes, and the Wikidata KB;
(iii) **Pseudo-question construction:** concatenate snippets per entity with temporal relations (*"before"*, *"after"* or *"during"*), and construct an interrogative sentence: a *pseudo-question*;
(iv) **Question rephrasing:** rephrase the pseudo-question into a natural question using a generative large language model (LLM).

We release a new benchmark, TIQ (**T**emporal **I**mplicit **Q**uestion), containing 10,000 questions and ground-truth answers with their original information snippets as supporting evidence. The benchmark has been used in our recent work [7], which focuses on faithful QA with temporally grounded explanations. The current paper focuses on the methodology for constructing the benchmark resource. The TIQ benchmark is available at **https://qa.mpi-inf.mpg.de/tiq**.

2 EXISTING BENCHMARKS

Benchmarks for temporal QA include [1, 4–6, 8, 9, 11, 18, 20, 22].

The majority of benchmarks have been released with a KB as dedicated knowledge source. TempQuestions [6] is one of the first benchmarks for temporal QA and collates temporal questions from existing general-purpose QA datasets. TimeQuestions [8] is an extension of TempQuestions with a total of 16K questions, covering more question types and temporal operators. TempQA-WD [11] is a subset of the original TempQuestions dataset, with answers and their ground-truth SPARQL queries for the Wikidata KB.

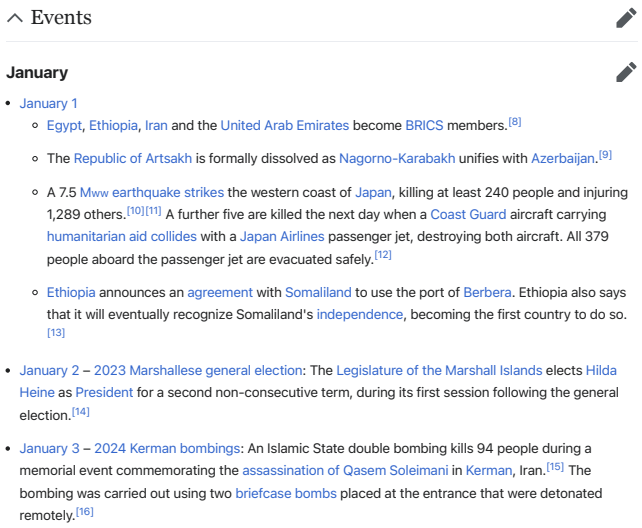
CronQuestions [18] is a larger template-based benchmark, and is generated from 30 seed templates based on 5 KB-relations, leading to a total of 410K questions. TempReason [20] has been specifically developed to test the capabilities of LLMs in answering temporal questions. Similar to the CronQuestions benchmark, questions are derived from templates based on 10 KB-relations. More recently, TempTabQA [5] has been released, providing questions formulated by crowd-workers based on Wikipedia infoboxes.

While TempQuestions, TimeQuestions, and TempQA-WD include implicit questions, their fraction is very low. CronQuestions and TempReason have a larger portion of implicit questions, but all questions are derived from a very small set of KB-relations (5 and 10, respectively) and hand-crafted rules, thus lacking diversity of formulations and intents. TempTabQA has a larger fraction of implicit questions but is also mono-source: questions are derived solely from Wikipedia infoboxes, which also leads to limited intents.

All existing benchmarks for temporal QA are tailored for a single source of answers. Table 1 compares characteristics of benchmarks, vis-a-vis our TIQ dataset.

3 CONSTRUCTION METHODOLOGY

We thus construct a new benchmark for temporal QA, with a focus on implicit questions, that is not targeted towards one specific knowledge source. While QA benchmarks are often constructed via crowdsourcing [17], this also comes with many pitfalls: annotation costs, sophisticated guidelines, all the way to workers merely invoking LLMs instead of providing natural questions themselves. Hence we propose an automated construction methodology instead.



Events

January

- January 1
  - Egypt, Ethiopia, Iran and the United Arab Emirates become BRICS members.[8]
  - The Republic of Artsakh is formally dissolved as Nagorno-Karabakh unifies with Azerbaijan.[9]
  - A 7.5 Mww earthquake strikes the western coast of Japan, killing at least 240 people and injuring 1,289 others.[10][11] A further five are killed the next day when a Coast Guard aircraft carrying humanitarian aid collides with a Japan Airlines passenger jet, destroying both aircraft. All 379 people aboard the passenger jet are evacuated safely.[12]
  - Ethiopia announces an agreement with Somaliland to use the port of Berbera. Ethiopia also says that it will eventually recognize Somaliland's independence, becoming the first country to do so.[13]
- January 2 – 2023 Marshallese general election: The Legislature of the Marshall Islands elects Hilda Heine as President for a second non-consecutive term, during its first session following the general election.[14]
- January 3 – 2024 Kerman bombings: An Islamic State double bombing kills 94 people during a memorial event commemorating the assassination of Qasem Soleimani in Kerman, Iran.[15] The bombing was carried out using two briefcase bombs placed at the entrance that were detonated remotely.[16]

Figure 1: Excerpt from the Wikipedia page for the year 2024.

Table 2: Prompt including demonstrations for rephrasing the pseudo-questions into natural questions.

Please rephrase the following input question into a more natural question.
Input: What album Sting (musician) was released, during, Sting award received German Radio Award? *Question*: which album was released by Sting when he won the German Radio Award?
Input: What human President of Bolivia was the second and most recent female president, after, president of Bolivia officeholder Evo Morales? *Question*: Which female president succeeded Evo Morales in Bolivia?
Input: What lake David Bowie He moved to Switzerland purchasing a chalet in the hills to the north of , during, David Bowie spouse Angela Bowie? *Question*: Close to which lake did David Bowie buy a chalet while he was married to Angela Bowie?
Input: What human Robert Motherwell spouse, during, Robert Motherwell He also edited Paalen 's collected essays Form and Sense as the first issue of Problems of Contemporary Art? *Question*: Who was Robert Motherwell's wife when he edited Paalen's collected essays Form and Sense?
Input: What historical country Independent State of Croatia the NDH government signed an agreement with which demarcated their borders, during, Independent State of Croatia? *Question*: At the time of the Independent State of Croatia, which country signed an agreement with the NDH government to demarcate their borders?
Input: What U-boat flotilla German submarine U-559 part of, before, German submarine U-559 She moved to the 29th U-boat Flotilla? *Question*: Which U-boat flotilla did the German submarine U-559 belong to before being transferred to the 29th U-boat Flotilla?
Input: What human UEFA chairperson, during, UEFA chairperson Sandor Barcs? *Question*: Who was the UEFA chairperson after Sandor Barcs?
Input: What human Netherlands head of government, during, Netherlands head of state Juliana of the Netherlands? *Question*: During Juliana of the Netherlands' time as queen, who was the prime minister in the Netherlands?

Key idea. A typical implicit question has two parts: the *main question* that specifies the information need disregarding time (e.g., *"Which team did Messi join"* for q_1), and the *implicit constraint* part that provides the actual temporal constraint (e.g., *"after Paris Saint-Germain"* for q_1). The key idea is to build the two parts from different pieces of evidence, denoted as *information snippets*. For example, the main part may originate from the text snippet *"Messi joined American club Inter Miami in July 2023"*, and the constraint may originate from a KB: *"Lionel Messi, member of sports team, Paris Saint-Germain F.C., start time August 2021, end time 30 June 2023"*.

One key property of the two information snippets is that they share the same *topic entity* (`Lionel Messi`), to ensure their thematic relatedness. Another criterion is that their temporal values (`July`

2023 and August 2021 – 30 June 2023) are compatible, and can be connected via a temporal relation (*"after"*). The answer (*"Inter Miami"*) comes from the information snippet underlying the main question. This snippet is then transformed into an interrogative form (*"What football team Messi joined"*), where the answer is replaced by its KB-type (*"football team"* in this case). The main question is connected with the implicit constraint via a temporal relation (*"after"*), for constructing an ungrammatical *pseudo-question* pq_1:

- pq_1: *What football team Messi joined, after, Lionel Messi member of sports team Paris Saint-Germain F.C.?*
- *answer*: Inter Miami
- *main*: *"Messi joined American club Inter Miami in July 2023"* (Text)
- *constraint*: *"Lionel Messi, member of sports team, Paris Saint-Germain F.C., start time August 2021, end time 30 June 2023"* (KB)

Finally, the pseudo-question is rephrased to obtain a natural question similar to q_1. This step also ensures lexical and syntactic diversity of the questions. The following sections will provide further detail.

3.1 Topic Entity Sampling

Year page retrieval. The benchmark construction starts with collecting significant events and their entities, as it is natural to ask questions about the temporal connection with striking events. Many such significant events are listed in the *year pages* in Wikipedia. These are dedicated Wikipedia pages that discuss the most important events in a year. Fig. 1 shows an excerpt from the year page for 2024 (https://en.wikipedia.org/wiki/2024). For some years, there are also monthly Wikipedia pages (e.g., https://en.wikipedia.org/wiki/November_1947), which allow for a finer granularity. We start with a specific range of years, 1801 - 2025 in our case, and identify all such year and month pages. Note that this range of years can be configured as required. From these Wikipedia pages, we collect all events and entities, and canonicalize them (to Wikipedia and Wikidata) using their href anchors. The result is a large set of candidate topic entities: we obtained 229,318 entities for the years 1801 - 2025.

Topic entity sampling. For each candidate topic entity, we look-up its *frequency* and *type* in the Wikidata KB via the CLOCQ API[1] [2]. This allows for controlling the distribution of long-tail vs. prominent topic entities by sampling topic entities of the desired frequency, and to control the domain coverage in questions by choosing topic entities of the desired types. The frequency of an entity is the number of KB facts with the entity. Entities with a frequency of less than 20 are considered long-tail, and entities with a frequency of more than 500 are considered prominent. The type of an entity is the most frequent type in the KB (in case there are multiple). Depending on the specific requirements, the set of topic entities can then be configured as desired (e.g., only long-tail entities, only entities related to movies, an equal ratio of prominent and long-tail entities, entities from a diverse set of domains,...).

For Tiq, we sampled a total of 10,000 topic entities. The amount of prominent and long-tail entities was balanced using roughly a 1:1 ratio. To ensure that topic entities come from a broad range of domains we restricted the fraction of a single entity type to 10%, i.e. there are no more than 1,000 entities of one type.

[1] https://clocq.mpi-inf.mpg.de

3.2 Information Snippet Retrieval

For the set of entities sampled from the previous stage, we retrieve relevant information snippets from (i) Wikipedia text, (ii) the respective Wikipedia infoboxes, and (iii) the Wikidata facts. The information sources or source combinations underlying the questions can again be configured (e.g., such that questions stem only from text).

Retrieving information snippets. For the Wikidata KB, we use the CLOCQ API to retrieve temporal KB facts [8] and linearize them into information snippets. This *verbalization* is implemented by concatenating the individual components (subject, predicate, object, qualifiers) of a KB fact with a comma [3, 13]. For example, we obtain information snippets such as *"Lionel Messi, date of birth, 24 June 1987"* or *"Lionel Messi, employer, FC Barcelona, start time, 2005, end time 2021"*. In some cases, such temporal information can be distributed in multiple facts with different relations (either inception and dissolved, or start time and end time). For example, there are two independent KB facts *"Commonwealth of Catalonia, inception, 6 April 1914"* and *"Commonwealth of Catalonia, dissolved, 1925"*. We merge these into a single information snippet: *"Commonwealth of Catalonia, inception, 6 April 1914, dissolved, 1925"*.

For Wikipedia pages we split text into sentences. Infoboxes are linearized similarly to KB facts, with each attribute-value pair making up one information snippet (e.g., *"Lionel Messi, Number, 10"*).

Annotating information snippets. We annotate entities in information snippets, which are then used as question entities or answer entities. For snippets from Wikipedia, we again utilize href anchors for entity linking and map them to Wikidata.

An important step is the identification and normalization of temporal expressions in the information snippets, to understand the temporal relationship between different snippets. Based on the granularity of temporal expressions, we categorize them into year, month, and date. For information snippets from KB, temporal expressions are already normalized as timestamps. For snippets from Wikipedia, temporal expressions are processed and normalized using regular expressions. We transform each temporal expression into a *temporal value*. This temporal value has a timestamp to indicate the start and end date, respectively. For example, the year *"2019"* becomes [2019-01-01,2019-12-31], the month *"December 2023"* becomes [2023-12-01,2023-12-31], the date *"6 April 1914"* becomes [1914-04-06,1914-04-06], and *"from 2001 to 2023"* becomes [2001-01-01,2023-12-31].

3.3 Pseudo-Question Construction

Sampling candidate snippets. The constraint of an implicit question is naturally a notable event, such as a striking global event (e.g., COVID-19) or an entity-specific event (e.g., winning an Academy Award). Thus, we restrict candidates for the constraint part to the more salient information of an entity:

(i) information snippets appearing on year pages (collected during topic entity sampling);
(ii) the first ten sentences in Wikipedia text that are typically providing salient information;
(iii) and information snippets from infoboxes and Wikidata that are generally notable as well.

As candidates for the main question part, we consider all information snippets for a topic entity from Wikipedia text, infoboxes, and Wikidata, irrespective of their salience. Besides the topic entity, the information snippets should mention at least one other entity, which becomes an answer candidate later. One restriction is that the answer should be time-sensitive, i.e. the answer should be different within different points in time, to ensure that the temporal constraint is actually relevant. For example, *"Messi's football team"* changes within different time intervals, but *"Messi's birth place"* would be the same for any temporal constraint.

Such sequential events are detected via semantic similarity, which is computed among information snippets of a topic entity. We use a SentenceTransformer[2] [16] to this end. For a snippet to be time-sensitive, there has to be another snippet with a distinct temporal value (otherwise it might be the same information expressed in different sources/snippets) that has a high semantic similarity. As threshold for semantic similarity we use 0.9 for snippets from KB and infoboxes, and 0.7 for snippets from text. We found this setting to perform best upon manual investigation. The threshold for the structured sources is higher, as there is little lexical divergence.

Collecting candidate pairs. For each candidate main part of a topic entity, candidate constraints need to satisfy two requirements: (i) they share at least one entity with the main part but not the answer entity, and (ii) their temporal values are comparable using either of the temporal relations *"during"*, *"after"*, or *"before"* (checked via rules). When using *"before"* or *"after"* as a connection, the gap in time should not be larger than 3 years. This avoids generating meaningless questions like *"For which football team did Messi play after he was born?"*.

Constructing pseudo-questions. For valid pairs, pseudo-questions are created, which are combinations of the snippets brought into interrogative forms. The question word *"what"* followed by the most frequent KB type of the answer entity is added to the beginning of the main question part (e.g., *"what football team"*). The mention of the answer entity is removed. Any temporal expressions and commas are removed from both the main and constraint parts, to avoid making the constraint explicit. Finally, the snippet for the main question, the temporal relation (*"during"*, *"after"*, or *"before"*), and the snippet for the constraint part are connected via a comma, as shown for pq_1 above. To avoid that questions become too fine-grained, pseudo-questions with more than 80 tokens or more than 10 entities are dropped. Note that we might obtain multiple pseudo-questions per topic entity.

In case there are different pseudo-questions with the same temporal relation and the same implicit constraint, but with highly similar main question parts (we apply the same semantic similarity computations and thresholding as above), this indicates that the two pseudo-questions express the same intent. Hence, the pseudo-questions are treated as a single instance, and their answers are grouped into a list of answer. An example is given below (there are two different snippets for the main part):

- pq_2: *What football team Messi joined, after, Lionel Messi member of sports team Barcelona?*
- *answer*: Inter Miami, Paris Saint-Germain
- *main*: *"Messi joined American club Inter Miami in July 2023"* (TEXT);
  "Lionel Messi, member of sports team, Paris Saint-Germain F.C., start time August 2021, end time 30 June 2023" (KB)
- *constraint*: *"Lionel Messi, member of sports team, October 2004 – April 2021, Barcelona"* (KB)

[2]https://huggingface.co/sentence-transformers/paraphrase-MiniLM-L6-v2

Table 3: Salient statistics of the TIQ benchmark.

Sources	Wikipedia text, infoboxes, and Wikidata
Questions	10,000 (train: 6,000, dev: 2,000, test: 2,000)
Avg. question length	17.96 words
Avg. no. of question entities	2.45
Unique topic entities covered	10,000
Long-tail topic entities covered	2,542 (with < 20 KB-facts)
Prominent topic entities covered	2,613 (with > 500 KB-facts)

The original information snippets are kept as supporting evidence, and for tracing how the (pseudo-)questions were obtained.

3.4 Question Rephrasing

Pseudo-questions are (by design) non-grammatical formulations, which need to be paraphrased to obtain a fluent and natural question. To this end, we utilize the language modeling capabilities of InstructGPT [14]. We hand-craft 8 demonstrations (pseudo-questions and their natural rephrasings) which are used for in-context learning, to generate the final question. The exact prompt is given in Table 2. For example, pq_2 is rephrased to *"Which clubs did Lionel Messi join after the FC Barcelona?"*. For diversity, we only sample one pseudo-question for each topic entity, for rephrasing.

3.5 Implementation Details

The previous sections provide the key steps of our benchmark construction methodology. In our implementation, we added some steps and mechanisms for efficiency, that are outlined in this section.

Partitioning the range of years. Users tend to be interested only in related events occurring close to each other in time. When considering a long time period (e.g., 200 years or so), the number of notable events is large, which leads to inefficient computations (e.g., when detecting related events). Hence, we split the initial year range (`1801 - 2025`) into smaller ranges of 50 years. After partitioning the initial year range, notable events within different smaller ranges are also separated. This means that valid candidates are lost. Thus, we choose the boundaries such that they overlap by 10 years (e.g., `1801 – 1860`, `1851 – 1910`,...) for each interval.

Sampling topic entities. After partitioning, we obtain a very large number of candidate topic entities for each time range. Retrieving information snippets, and constructing pseudo-questions for all of these topic entities is infeasible. We thus iteratively obtain a smaller random sample of entities (e.g., 100), and then run the construction pipeline until we obtain the desired number of pseudo-questions. The sampled pseudo-questions are then rephrased via the LLM. After sampling and rephrasing, we filter out noisy questions (e.g., questions including dates or very long questions) to ensure that questions in the benchmark are implicit and readable. Since we obtain multiple pseudo-questions for each topic entity (the average number of pseudo-questions per entity is 27.91), the sampling and rephrasing process is conducted iteratively until we reach the target number of questions for the benchmark (10,000 in our case).

4 TIQ BENCHMARK

4.1 Characteristics

The TIQ dataset has 10,000 questions, and is split into train (6,000), dev (2,000), and test sets (2,000). Table 3 provides salient statistics of the benchmark. It is available at **https://qa.mpi-inf.mpg.de/tiq**.

Metadata. The TIQ dataset includes metadata that can be useful for training and evaluating (components of) temporal QA systems:

- the gold answers as text, linked to Wikipedia and linked to Wikidata,
- the information snippets grounding the question,
- the sources these were obtained from,
- the normalized temporal values expressed in the information snippets,
- the temporal relation, and
- the topic entity and question entities detected in the snippets.

Table 4 shows example questions in TIQ.

4.2 Analysis

Topic Entities. In TIQ, each question is derived from a unique topic entity. There are 2,542 long-tail entities and 2,613 prominent entities in the benchmark (Table 3). There are 1,100 different (KB) types of topic entities, covering top-level domains like "human", "city", "sports", "country", "business", "university", "politics", "film/TV series", "music", "organization/company", or "museum".

Question properties. The distributions of the number of question entities and the question length are shown in Fig. 3. The distribution of question words in the benchmark (e.g., *"What"*, *"Who"*, *"Which"*) and the distribution of temporal relation types are shown in Fig. 4.

Answers properties. The number of answers for each question varies from 1 to 10, with the average being 1.07. The number of different answer KB types is 1,143, and the top-level domain coverage is similar to the topic entities ("human", "city", "sports", ...).

Information sources. There are 20,624 information snippets grounding the questions in the benchmark. The number of questions derived from each information source is shown in Fig. 2. The distributions for the main question parts and constraint parts are shown as well. Overall, the information sources are fairly balanced.

Temporal values. We analyzed the granularity of temporal expressions in the information snippets. The number of years is 12,094, there are 538 months, and 7,992 dates. Months are naturally a bit rarer than years and dates. For the main part, the number of years, months and dates is 6,670, 204 and 3,798. For the constraint part, the number of years, months and dates is 5,424, 334 and 4,196.

4.3 Correctness

We randomly sampled 100 questions from the benchmark and manually inspected to check if they preserve the intent of the original pseudo-questions. The portion of the questions preserving the original meaning is 82%. The reasons for failures include four aspects:(i) main predicate wrong (8%), (ii) constraint wrong (9%), (iii) answer type wrong (6%), and (iv) temporal conjunction wrong (2%). The number of error cases adds up to 25% (> 18%) because some questions fail in multiple aspects. For example, for two questions, the LLM switched their main and constraint parts resulting in errors in both parts of the rephrased questions.

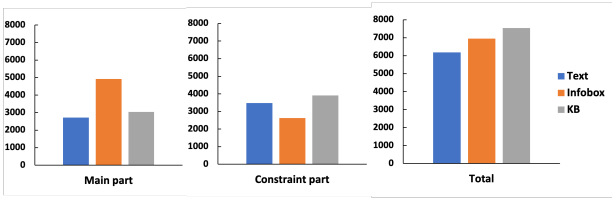


Figure 2: Integration of the different information sources.

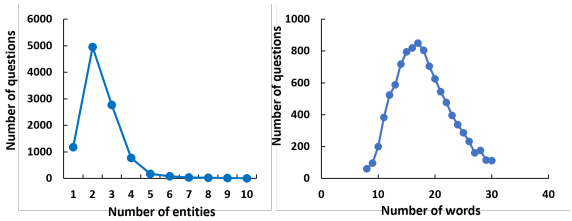


Figure 3: Distribution of number of entities / question length.

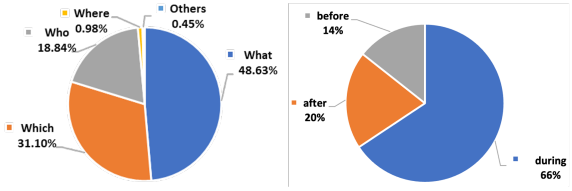


Figure 4: Distribution of question words / temporal relations.

5 EXPERIMENTS

5.1 Experimental setup

Metrics. We measure the following metrics: precision at 1 (P@1), mean reciprocal rank (MRR), and hit at 5 (Hit@5) [17]. Metrics are averaged over all questions.

Methods. We run a wide range of methods on TIQ, to understand which types of methods perform best on the benchmark:

- **Generative LLMs**. We run **InstructGPT** [14] ("text-davinci-003") and **GPT-4** [12] ("gpt-4") using the OpenAI API[3]. The following prompt showed the best performance among our candidates: *"Please answer the following question by providing the crisp answer entity, date, year, or number."*. As we only obtain a single generated answer, MRR and Hit@5 are not applicable for both LLMs. We compute P@1 by checking whether the generated answer string matches with the label or any alias of the ground-truth answer (P@1 is 1 if so, and 0 otherwise).
- **Heterogeneous QA methods**. As TIQ is constructed from heterogeneous sources, we run a set of recent general-purpose methods operating over heterogeneous sources: **Uniqorn** [15], **UniK-QA** [13], and **Explaignn** [3].
- **Temporal QA methods**. Finally, we run the state-of-the-art for temporal QA: **Exaqt** [8], and **Faith** [7].

Configuration. Wikidata [21] is used as the KB (whenever applicable). We use Wikipedia text, tables and infoboxes as additional sources for methods operating over heterogeneous sources. All methods are trained on the benchmark (except for the GPT-models).

[3]https://platform.openai.com

Table 4: Example questions from the TIQ benchmark, including the information snippets they are derived from.

Topic entity	`Clarence Andrew Cannon`
Question	*What was Clarence Andrew Cannon's occupation before becoming a lawyer?*
Answer	teacher
Information snippet - Main	*"Clarence Andrew Cannon, occupation, teacher, start time, 1904, end time, 1908"* (KB)
Information snippet - Constraint	*"Clarence Cannon, He earned an LL.B. and joined the bar in 1908."* (TEXT)
Topic entity	`Robert Bosch GmbH`
Question	*Who was the chief executive officer at Robert Bosch GmbH before revenue reached €78.74 billion?*
Answer	Volkmar Denner
Information snippet - Main	*"Robert Bosch GmbH, chief executive officer, Volkmar Denner, start time, 2012, end time, 2021"* (KB)
Information snippet - Constraint	*"Robert Bosch GmbH, Revenue, € 78.74 billion (2021)"* (INFOBOX)
Topic entity	`Carlos Alberto Torres`
Question	*Which national football team did Carlos Alberto Torres manage before joining Flamengo?*
Answer	Oman national football team
Information snippet - Main	*"Carlos Alberto Torres, Managerial career, 2000–2001, Oman"* (INFOBOX)
Information snippet - Constraint	*"Carlos Alberto Torres, Managerial career, 2001–2002, Flamengo"* (INFOBOX)
Topic entity	`Alan Page`
Question	*What hall of fame did Alan Page become a member of while serving as Associate Justice of the Minnesota Supreme Court?*
Answer	College Football Hall of Fame
Information snippet - Main	*"Alan Page, In 1993, he was inducted into the College Football Hall of Fame."* (TEXT)
Information snippet - Constraint	*"Alan Page, Associate Justice of the Minnesota Supreme Court, In office January 4, 1993 – August 31, 2015"* (INFOBOX)

Table 5: Main results comparing a range of methods (LLMs, temporal QA, heterogeneous QA) on the TIQ dataset.

Method	P@1	MRR	Hit@5
INSTRUCTGPT [14]	0.237	n/a	n/a
GPT-4 [12]	0.236	n/a	n/a
UNIQORN [15]	0.236	0.255	0.277
UNIK-QA [13]	0.425	0.480	0.540
EXPLAIGNN [3]	0.446	0.584	0.765
EXAQT [8]	0.232	0.378	0.587
FAITH [7]	0.491	0.603	0.752

5.2 Experimental Results

The experimental results are shown in Table 5.

TIQ poses a challenge for existing methods. None of the existing methods from the literature is able to answer at least half of the questions in the benchmark. This finding demonstrates that TIQ is a challenging dataset, going beyond the capabilities of the state-of-the-art. Notably, this includes recent GPT-models indicating that the implicit questions in TIQ require dedicated answering mechanisms beyond the general-purpose language models.

Anecdotal failure cases. We identified that for 13.50% of the questions none of the methods was able to compute the correct answer. Examples include (i) *"What was Harry Reid's position before he became a United States Senator from Nevada?"*, (ii) *"When Pegasus was initially released, what exploit was discovered in the Unix-like spyware?"*, and (iii) *"Before playing for CSKA Moscow, which basketball team was Mirsad Türkcan a part of?"*.

6 CONCLUSION

This work introduced the TIQ benchmark for temporal QA with implicit constraints about time points or time spans. Prior QA benchmarks do not adequately cover this challenging segment of user questions. A key point in constructing the benchmark is that we can systematically control and vary important factors like topical diversity, formulation style and question difficulty (e.g., head entities vs. long tail). We hope that the TIQ benchmark can contribute to advancing future research on temporal QA.

Acknowledgements. Zhen Jia was supported by NSFC (Grant No.62276215 and No.62272398).

REFERENCES

[1] Wenhu Chen, Xinyi Wang, and William Yang Wang. 2021. A Dataset for Answering Time-Sensitive Questions. In *NeurIPS*.
[2] Philipp Christmann, Rishiraj Saha Roy, and Gerhard Weikum. 2022. Beyond NED: Fast and Effective Search Space Reduction for Complex Question Answering over Knowledge Bases. In *WSDM*.
[3] Philipp Christmann, Rishiraj Saha Roy, and Gerhard Weikum. 2023. Explainable Conversational Question Answering over Heterogeneous Sources via Iterative Graph Neural Networks. In *SIGIR*.
[4] Bhuwan Dhingra, Jeremy R Cole, et al. 2022. Time-Aware Language Models as Temporal Knowledge Bases. In *TACL*.
[5] Vivek Gupta, Pranshu Kandoi, et al. 2023. TempTabQA: Temporal Question Answering for Semi-Structured Tables. In *EMNLP*.
[6] Zhen Jia, Abdalghani Abujabal, et al. 2018. TempQuestions: A Benchmark for Temporal Question Answering. In *HQA@WWW*.
[7] Zhen Jia, Philipp Christmann, and Gerhard Weikum. 2024. Faithful Temporal Question Answering over Heterogeneous Sources. In *WWW*.
[8] Zhen Jia, Soumajit Pramanik, et al. 2021. Complex Temporal Question Answering on Knowledge Graphs. In *CIKM*.
[9] Adam Liska, Tomas Kocisky, et al. 2022. Streamingqa: A benchmark for adaptation to new knowledge over time in question answering models. In *ICML*.
[10] Costas Mavromatis, Prasanna Lakkur Subramanyam, et al. 2022. TempoQR: Temporal Question Reasoning over Knowledge Graphs. In *AAAI*.
[11] Sumit Neelam, Udit Sharma, et al. 2022. SYGMA: System for Generalizable Modular Question Answering over Knowledge Bases. In *EMNLP*.
[12] OpenAI. 2023. GPT-4 Technical Report. In *arXiv*.
[13] Barlas Oğuz, Xilun Chen, et al. 2022. UniK-QA: Unified Representations of Structured and Unstructured Knowledge for Open-Domain Question Answering. In *NAACL-HLT*.
[14] Long Ouyang, Jeffrey Wu, et al. 2022. Training Language Models to Follow Instructions with Human Feedback. In *NeurIPS*.
[15] Soumajit Pramanik, Jesujoba Alabi, et al. 2021. UNIQORN: Unified Question Answering over RDF Knowledge Graphs and Natural Language Text. In *arXiv*.
[16] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *EMNLP*.
[17] Rishiraj Saha Roy and Avishek Anand. 2022. *Question Answering for the Curated Web: Tasks and Methods in QA over Knowledge Bases and Text Collections*. Springer.
[18] Apoorv Saxena, Soumen Chakrabarti, and Partha Talukdar. 2021. Question Answering Over Temporal Knowledge Graphs. In *ACL*.
[19] Jannik Strötgen and Michael Gertz. 2016. *Domain-sensitive temporal tagging*. Springer.
[20] Qingyu Tan, Hwee Tou Ng, and Lidong Bing. 2023. Towards Benchmarking and Improving the Temporal Reasoning Capability of Large Language Models. In *ACL*.
[21] Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: A Free Collaborative Knowledgebase. *CACM* (2014).
[22] Michael JQ Zhang and Eunsol Choi. 2021. SituatedQA: Incorporating extra-linguistic contexts into QA. In *EMNLP*.