data

MDPI

Data Descriptor

TKGQA Dataset: Using Question Answering to Guide and Validate the Evolution of Temporal Knowledge Graph

Ryan Ong [1,*], Jiahao Sun [1,2], Ovidiu Șerban [1] and Yi-Ke Guo [1]

[1] Department of Engineering, Imperial College London, London SW7 2BX, UK
[2] Royal Bank of Canada, London EC2 4AA, UK
* Correspondence: cmo18@imperial.ac.uk

Abstract: Temporal knowledge graphs can be used to represent the current state of the world and, as daily events happen, the need to update the temporal knowledge graph, in order to stay consistent with the state of the world, becomes very important. However, there is currently no reliable method to accurately validate the update and evolution of knowledge graphs. There has been a recent development in text summarisation, whereby question answering is used to both guide and fact-check summarisation quality. The exact process can be applied to the temporal knowledge graph update process. To the best of our knowledge, there is currently no dataset that connects temporal knowledge graphs with documents with question–answer pairs. In this paper, we proposed the TKGQA dataset, consisting of over 5000 financial news documents related to M&A. Each document has extracted facts, question–answer pairs, and before and after temporal knowledge graphs, to highlight the state of temporal knowledge and any changes caused by the facts extracted from the document. As we parse through each document, we use question–answering to check and guide the update process of the temporal knowledge graph.

Dataset: https://doi.org/10.17605/OSF.IO/XQWA4

Keywords: temporal knowledge graph; question–answering; knowledge graph; entity dynamic; event knowledge graph; mergers and acquisitions; finance

Citation: Ong, R.; Sun, J.; Șerban, O.; Guo, Y.-K. TKGQA Dataset: Using Question Answering to Guide and Validate the Evolution of Temporal Knowledge Graph. *Data* **2023**, *8*, 61. https://doi.org/10.3390/data8030061

Academic Editor: Juanle Wang

Received: 29 October 2022
Revised: 23 January 2023
Accepted: 12 March 2023
Published: 14 March 2023

1. Introduction

A knowledge graph represents the world through structural facts, consisting of entities and relationships. Entities are real-world objects, such as people, companies, countries, etc., and relationships capture the relation between these real-world objects. In a knowledge graph, a fact is represented by a triplet of head entity, relation, tail entity; for example, apple, success_acquire, Netflix. Most knowledge graph research has focused on static knowledge graphs, where facts remain unchanged over time. However, this is not a realistic real-world environment. Facts can change, which means that some facts that are true today might no longer be true in the future. In order to capture the changes in the validity of facts, we need to update and evolve the knowledge graph, which falls under the temporal knowledge graph research category.

Unlike static knowledge graphs, facts in temporal knowledge graphs are represented by quads of head entity, relation, tail entity, and validity period, where the validity period means the period during which the fact is valid. Research in temporal knowledge graphs can be split into the following four categories [1]: temporal information embedding, entity dynamics, temporal relational dependency, and temporal logical reasoning.

Many daily events change the state of the world, and our objective is to capture the impact of these real-world events on entities and relations (via news articles). We want to

update the temporal knowledge graph accordingly, so that knowledge-aware applications, built on the knowledge graphs, perform accurately with up-to-date information. This area of research falls under entity dynamics. In temporal knowledge embedding, or, commonly, for the task of link prediction, there are easily accessible standardised datasets, such as YAGO15K, WIKIDATA, ICEWS, and GDELT [2,3]. However, this is not the case in entity dynamics, in which there are no easily accessible standardised datasets. For example, the TextWorld KG dataset [4] was introduced to build dynamic knowledge graphs from text-based games. However, given the fixed nature of games, the future state of the temporal knowledge graph is known with high confidence, which does not accurately represent the real-world environment. The dynamic knowledge graph was built using procedural text [5] (PROPARA dataset) to track the evolving states of entities. Contextual temporal profiles [6] were used to detect state changes in entities, and NBA transactions [7] was created using NBA data to capture player trades between different basketball teams.

In text summarisation, research work has started using question–answering rewards to guide and fact-check summarisation [8–11]. The macro theme is to have a set of questions (either human-generated or model-generated) and answers, paired, with each original document, so that we can perform question–answering on the generated summaries to see if the generated answers are similar to the ground-truth answers. We strongly believe that the question–answering rewards can be used to guide and fact check the update process of temporal knowledge graphs. Additionally, it can guide the knowledge extraction process to train the model to better extract entities and relations from news articles. Unfortunately, to the best of our knowledge, no dataset focuses on the complete end-to-end pipeline, from NER extraction to the updating of a temporal knowledge graph, by means of question–answering.

Question–answering for temporal knowledge graphs is a relatively new research area, with many Q&A datasets failing to realistically represent real-world settings. For example, TORQUE [12] is a temporal Q&A dataset that consists of query questions, context, and multiple-choice questions, with answers. This is not realistic, since, in real-world settings, the model is expected to generate answers out of hundreds of thousands of entities, with little to no context. There are other limitations of existing temporal Q&A datasets [13,14]: (a) they are relatively small datasets and, more importantly, (b) the temporal questions are simple and are applied to a non-temporal knowledge graph [15,16]. In 2021, CRONQUESTIONS [17] provided a good dataset contribution, consisting of simple and complex temporal questions and a temporal knowledge graph. However, it still does not accurately represent real-world settings, since it assumes a fixed temporal knowledge graph, and the question–answering is not connected to accessible documents. A dataset that closely resembles the real-world environment and that enables question–answering rewards should have the following conditions: (1) a temporal knowledge graph that includes all the facts and changes caused by documents; (2) documents with extracted tuples, and (3) documents paired with questions and answers (within the document or on external data).

To address these conditions, we proposed a new standardised dataset, the TKGQA dataset, which consists of over 5000 financial news articles regarding the M&A. Each document is paired with extracted tuples (human-extracted and model-predicted) and relevant questions and answers. There are also two temporal knowledge graphs for each document; capturing the state of the temporal knowledge graph before and after the latest facts (extracted tuples). This mimics the real-world environment, as we have a dynamic knowledge graph that is updated with frequent news articles, question–answering is used to check if the update process covers both explicit and implicit changes.

The rest of the paper is structured as follows. Section 2 describes, in detail, the overall data structure of the TKGQA dataset, its data records and key statistics, as well as its guide to access EDA codes and the data repository. Section 3 outlines the overall process in creating the TKGQA dataset, broken down into four main steps of data collection, facts extraction, TKG generation, and Q&A generation. Section 4 describes the methods used to

validate the quality and reliability of the TKGQA dataset. Finally, in Section 5, we describe primary and secondary use cases of the TKGQA dataset.

2. Data Description

We archived seven types of data records with Open Science Framework (OSF) [18], accessed on 24 October 2022, at https://doi.org/10.17605/OSF.IO/XQWA4. The main data record is the main dataset, which contains 5721 documents related to the stages of M&A deals. Each document has extracted tuples, made up of entities, relations, and date as well as a list of general, head entity, and tail entity questions related to the M&A deal, to be used to assess the quality of the update process. Additionally, each document is paired with two TKGs, illustrating the state of TKG before and after ingesting the document. Figure 1 showcases the attributes we have for each document in the TKGQA dataset. The temporal knowledge graph consists of tuples of head entity, relation, tail entity, start date, and end date. The start date and end date of facts represents the validity of the facts and as we parse through each document in the main dataset, and depending on the M&A deal, we update the temporal knowledge graph accordingly. The snapshots of the temporal knowledge graph can easily be queried using the time period.

The second type of data record is the entities list that contains all the entities that we extracted from the sentences and that we were able to link to WIKIDATA. For each entity, we normalised the names, removed any duplications, and retrieved more information and attributes from WIKIDATA.

The third, fourth, and fifth types of data record are the entities, relations, and timestamp ids. Each snapshot of the TKG has its own id files, since each has its own set of entities, relations, and timestamps, and since new documents might introduce new entities/relations.

Lastly, we included the documents and the mappings of the document ids to their respective URLs, to provide references for each article at the dataset level.

Each data record is saved in either .csv or .pickle format. The statistics on the TKGQA dataset, extracted entities, and the temporal knowledge graphs are shown in Tables 1–3. For more information on the EDA work presented in this paper, please visit https://github.com/RyanOngAI/m-a_temporal_knowledge_graph_qa, accessible on 24 October 2022. We have included few data samples of the TKGQA dataset in Appendix A.

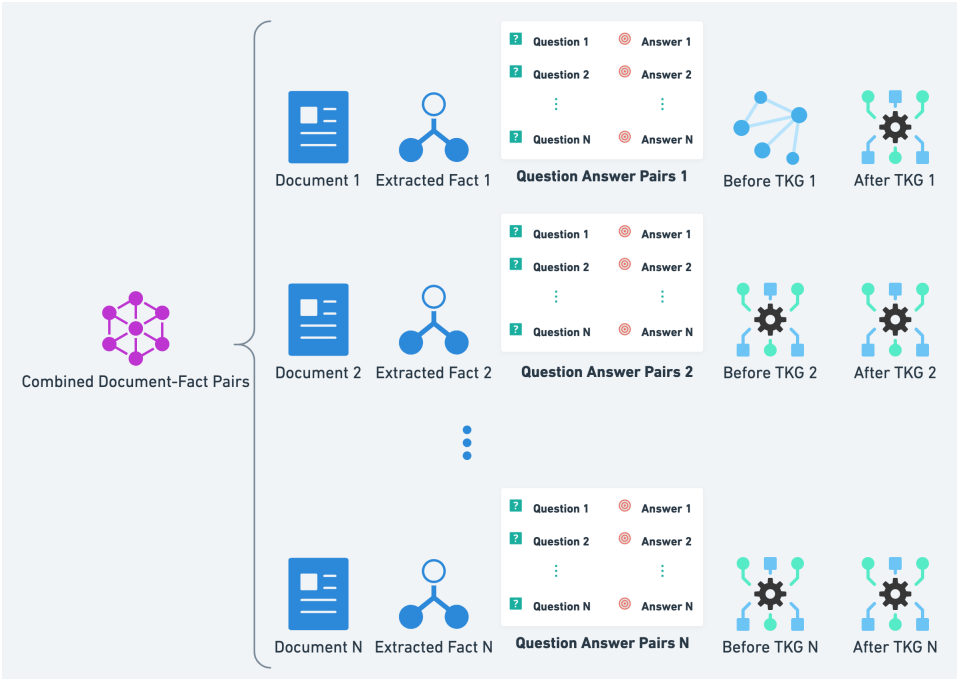

Figure 1. TKGQA Dataset.

Table 1. Statistics on TKGQA dataset.

TKGQA Dataset Documents	Count
Total number of documents	5721
Total number of general questions per document	3
Total number of documents having head questions	2425
Average number of questions for head entity	15.32
Average number of questions for tail entity	1.014
Maximum number of questions for head entity	289
Maximum number of questions for tail entity	195
Total number of extracted entities	2527
Average deal per entity	1.83
Maximum deal per entity	74

Table 2. Top Ten Industries for extracted entities.

Industries	Count
financial service (Q837171)	179
telecommunications industry (Q25245117)	84
retail (Q126793)	75
pharmaceutical industry (Q507443)	69
petroleum industry (Q862571)	65
software industry (Q880371)	65
Finanzwesen (Q1416657)	61
banking industry (Q806718)	56
automotive industry (Q190117)	49
video game industry (Q941594)	49

Table 3. Statistics on Temporal Knowledge Graph Snapshots.

Temporal Knowledge Graph	Count
Total number of TKG Snapshots	5722
Total number of facts (last state)	21,725
Total original facts	8319
Total added facts	13,406
Total modified facts	1605
Total number of unique entities	14,756
Total number of unique relations	13

3. Methods

3.1. Data Creation

The TKGQA dataset consists of financial news articles regarding mergers and acquisitions, spanning from January, 2018, through to June, 2021. Each news article has three main data points: (1) the extracted tuples (from the article), (2) questions and answers related to the article, and (3) two snapshots of the temporal knowledge graph, covering the state of the knowledge graph before and after the article.

The data creation process can be broken down into four main steps:

1. Data Collection
2. Facts Extraction
3. TKG Generation
4. Q&A Generation

Figure 2 illustrates the entire workflow of the data collection and facts extraction process. In data collection, we used 8 keywords to scrape news articles from January 2018, to June 2021, using commercial News API (https://newsapi.org/docs); merger (15,669), merge (3881), merging (658), merged (409), acquisition (16,809), acquire (11,655), acquiring

(975), and acquired (3630), totalling 53,686 articles. We manually removed any articles that were irrelevant to mergers and acquisitions.

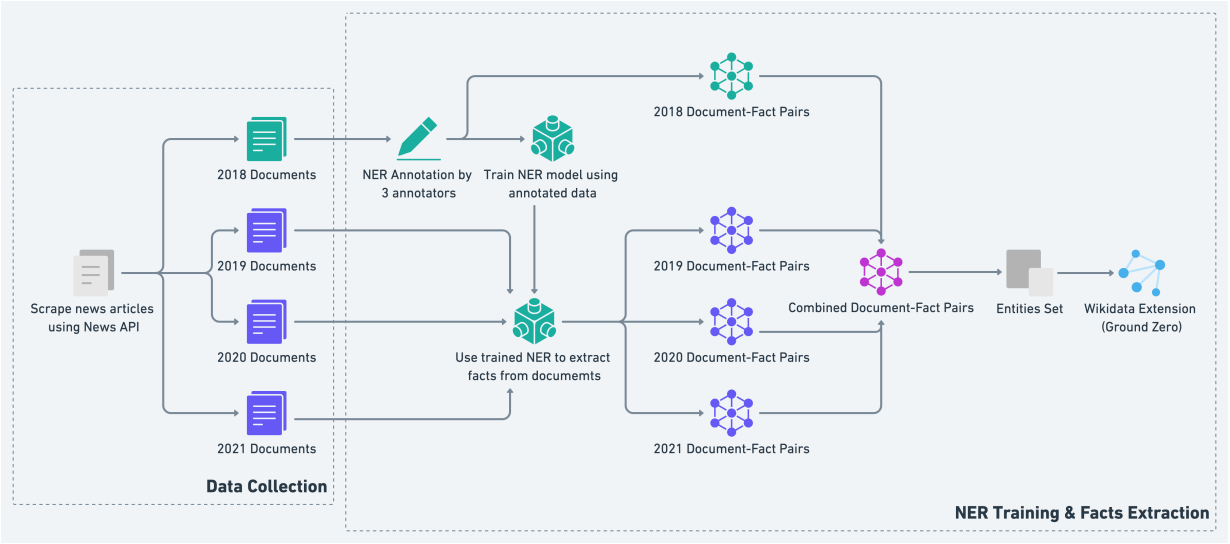

Figure 2. Data Collection and Facts Extraction.

In the facts extraction pipeline, our goal was to extract both entities (companies) and relations (stages of M&A) from news articles. In order to achieve this, we trained a specific NER to detect beyond the general organisation tag, as well as to detect the different stages of M&A. Since news articles contain much noisy information, we decided to split the articles into sentences to better enable us to train our NER model. We split all the news articles into sentences, giving us 155,713 raw sentences. We decided to use sentences from 2018 to train our NER model, which gave us a raw training set of 47,704 sentences. We removed any irrelevant sentences that did not contain information on M&A deals and, for simplicity, we treated relations as "entities", so that we only needed to train a single NER model to extract both entities and relations. In total, we had 3 entities and 4 relations:

- Bidder—the company looking to merge or acquire another company
- Target—the company being merged/acquired
- Considering—early stage of discussion/talks, pre-approval
- Expecting—anything that signifies high probability of the deal going through
- Success—deals completed, agreed/signed to acquire, merged, acquired, entered/reached
- Terminated—deals cancelled, refused
- Org—general companies that are not part of the M&A deal

In order to train our custom NER, we hired three annotators to annotate the 2018 sentences on Doccano [19], an open-source text annotation tool for humans, using the predefined entities and relations above. Specifically, for each sentence, we extracted the start and end indices of all detected entities and used them to fine-tune a spaCy NER model, which is a modified CNN architecture with Bloom filter. We used the trained NER to extract entities and relations (facts) in sentences from January, 2019, to June, 2021. We used the date of the articles as a proxy for the validity of facts, since financial news articles are often published close to the time at which the M&A deals happen. Additionally, we removed any predictions where there was a mismatch between the predicted labels and the keywords used to extract the sentences. An example of a mismatch would be if the sentence was extracted using the keyword "acquired" but our trained NER predicted a merger-related relation.

We tidied up the predictions (facts extraction), by removing duplicated and similarly named entities, and consolidated the final set of extracted entities. For each entity in our entities set, we extracted additional information from WIKIDATA, that were relevant to

M&A deals, to monitor both explicit and implicit changes. We extracted the following WIKIDATA attributes for both our question answering (QA) templates and exploratory data analysis (EDA) tasks:

- owner of (P1830)—QA
- subsidiary (P355)—QA
- owned by (P127)—QA
- business division (P199)—QA
- board member (P3320)—QA
- industry (P452)—EDA
- founded by (P112)—EDA
- inception (P571)—EDA
- stock exchange (P414)—EDA
- country (P17)—EDA
- part of (P361)—EDA

In TKG generation, we initialised the starting state of the TKG using the extracted WIKIDATA attributes and values, because, when M&A deals go through, there are both explicit and implicit changes that happen within companies. The explicit changes are reflected in the news articles, but the implicit changes are usually not mentioned in the news articles and, as such, we decided to use the attributes (and the changes in these attributes) from WIKIDATA to capture the implicit effect.

As we parsed each document, we updated the temporal knowledge graph accordingly. This involved using a rule-based algorithm to add/adjust the facts depending on the stage of the M&A deal (relations). Since implicit changes only occur when the M&A deal is successful, in most cases, we only added the explicit information (a single fact about the latest stage of the deal) to our temporal knowledge graph. When the M&A deal was successful (success relation), we added/adjusted facts, depending on the attributes. The attributes "owner of (P1830)", "subsidiary (P355)", and "business division (P199)" represent the assets of a company and, as such, in a successful acquisition deal, we would add facts that connected the acquirer (head entity) to the acquiree's (tail entity) assets. The attributes "board member (P3320)" and "owned by (P127)" represent the shareholder of a company and, as such, we would add an end date to all the "board members" and "owned by" of the acquiree, if the company is acquired. We used the date of the articles as a proxy for the start date and end date of new facts, since financial news articles are usually published close to the time at which the M&A deal happens. For a successful merger deal, we only added a single fact of the explicit information to connect the two companies together, since we viewed a merger as a partnership, rather than a transferring of assets. Once we parsed through all the documents, we had different snapshots and the final state of the temporal knowledge graph was acquired.

Lastly, for Q&A generation, for each article we created templates (as shown in Table 4) that used WIKIDATA attributes to generate a set of questions and answers for both before and after the parsing of the document. These questions and answers were used to check whether the temporal knowledge graph was updated accurately. Figure 3 showcase the entire workflow from TKG to Q&A generation.

Table 4. Templates used to generate questions and answers for documents. [**HEAD**], [**TAIL**] represents entities. [**SUBJECT ENTITY**] represents entity values from Wikidata.

Attributes	Template
owner of (P1830)	Who owns **[SUBJECT ENTITY]** before the latest status of the deal between **[HEAD]** and **[TAIL]**?
subsidiary (P355)	Who does **[SUBJECT ENTITY]** belong to before the latest status of the deal between **[HEAD]** and **[TAIL]**?
owned by (P127)	Who owns **[HEAD]** after the latest status of the deal between **[HEAD]** and **[TAIL]**?
business division (P199)	Who does **[SUBJECT ENTITY]** belong to after the latest status of the deal between **[HEAD]** and **[TAIL]**?
board member (P3320)	Who has influence over **[TAIL]** after the latest status of the deal between **[HEAD]** and **[TAIL]**?

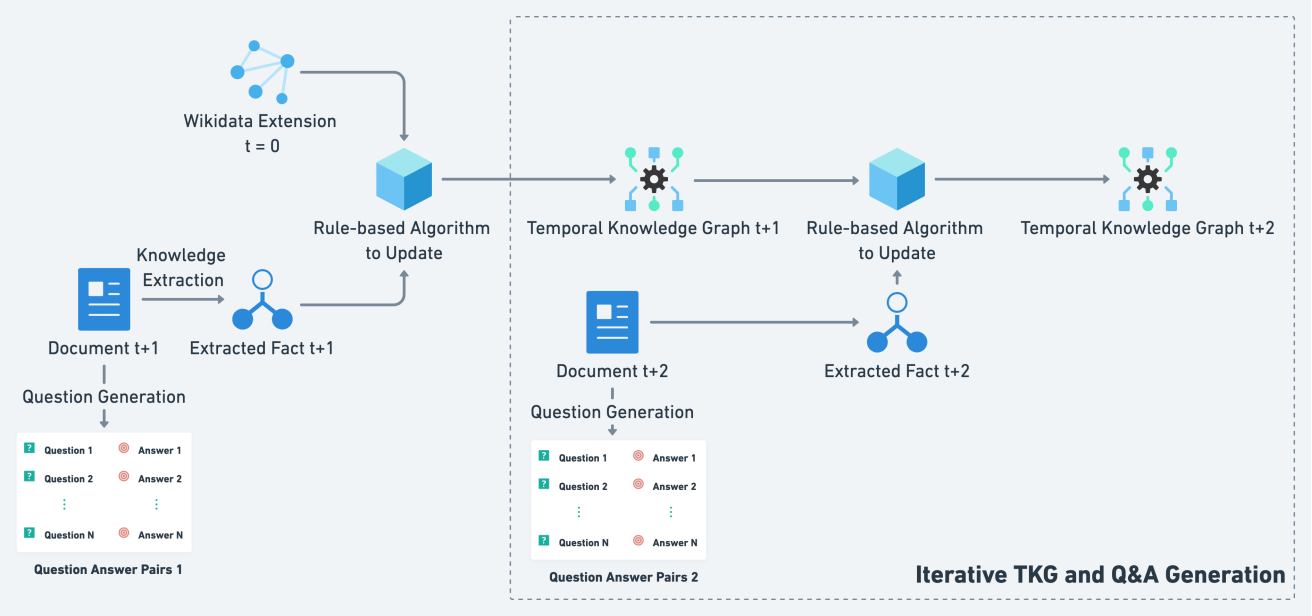

Figure 3. TKG and Q&A Generation.

3.2. Evaluation Metrics

To evaluate the agreements between our annotators for the NER annotations, we used the popular inter-annotator agreement (IAA) measured by the Cohen's Kappa score. Cohen's Kappa [20] is commonly used to measure agreements between annotators, and is known to be a more reliable measurement than percentage agreement in evaluating the quality of annotations. Since each annotator annotated sentences at the token-level for the NER training data, Cohen's kappa compared the token-level annotations between two annotators and computed a score to represent the overall annotation agreements between them. Cohen's kappa ranges between 0 and +1. We had three types of Cohen's kappa computations: (1) between annotators; (2) between generated gold labels and annotators; and (3) between authors' labels and annotators. For 3), we (the authors) manually labelled 1200 sentences, consisting of 400 sentences from each annotator's annotation set, so that we could compare our annotations against the annotators' annotations to further assess the quality of the annotators' annotations.

The gold labels were computed using simple rule-based algorithms to combine the annotations between two and three annotators. For annotations where we had three annotators, we simply used majority voting to decide what the final gold labels should be. With the gold labels we could compute IAA scores between each annotator and the gold labels. For annotations where we only had two annotators, we had two simple rules. Firstly, we had a bias towards entities tags, meaning that, if, for example, for the same token, we had an O tag (non-entity) and a Bidder tag, we would be biased towards choosing the Bidder tag as the final gold label. Secondly, in instances where both annotations for the same token were entities tags, we would be biased towards the annotator with a higher IAA score computed using the gold labels.

4. Technical Validation

4.1. Inter-Annotator Agreement (IAA)

As mentioned above, to assess the quality of annotations, we computed the IAA scores between the annotators. A high IAA score meant there was a high agreement level between the annotators, signalling that the annotations were reliable for use in training our NER models.

In addition to computing IAA scores between annotators, we also computed the IAA scores between our gold labels and authors' labels to compare the overall agreement between the general consensus of our annotators (gold labels) and our own labelling.

The IAA scores were computed at the token level and, as such, we included IAA scores computed with and without O tags (non-entities tokens). Scores with O tags were always higher than scores without O tags, since there were many O tags tokens within a given sentence and, as such, any match between annotators on the O tags was considered to be correct. All the IAA results are shown in Table 5.

Table 5. All the IAA score computations. The bolded IAA scores represent the best achieving score within each comparison category, i.e., Annotator vs. Annotator with O tags, Annotator vs. Annotator without O tags, etc.

Computations	IAA (Cohen Kappa Score)
Annotator 1 vs. Annotator 2 (with O tags)	0.75
Annotator 1 vs. Annotator 3 (with O tags)	0.75
Annotator 2 vs. Annotator 3 (with O tags)	**0.80**
Annotator 1 vs. Annotator 2 (without O tags)	0.63
Annotator 1 vs. Annotator 3 (without O tags)	0.61
Annotator 2 vs. Annotator 3 (without O tags)	**0.69**
Authors' Labels vs. Annotator 1	0.75
Authors' Labels vs. Annotator 2	0.77
Authors' Labels vs. Annotator 3	**0.80**
Authors' Labels vs. NER Algorithm (all annotators)	**0.66**
Authors' Labels vs. NER Algorithm (annotator 1)	0.64
Authors' Labels vs. NER Algorithm (annotator 2)	0.67
Authors' Labels vs. NER Algorithm (annotator 3)	0.66
Gold Labels vs. Annotator 1 (with O tags)	0.80
Gold Labels vs. Annotator 2 (with O tags)	**0.84**
Gold Labels vs. Annotator 3 (with O tags)	**0.84**
Gold Labels vs. Annotator 1 (without O tags)	0.75
Gold Labels vs. Annotator 2 (without O tags)	**0.83**
Gold Labels vs. Annotator 3 (without O tags)	0.82
Gold Labels vs. Authors' Labels	**0.77**
Gold Labels vs. NER Algorithm	**0.65**

From the results, it was apparent that annotators 2 and 3 had the highest agreements in both IAA with O tags (0.80) and IAA without O tags (0.69). To assess the quality of an individual's annotations, we computed the IAA between authors' labels and the annotators and the results showed that annotator 3 had the highest quality of annotations, followed by annotator 2, and then annotator 1. This was useful for when we computed gold labels between two annotators, as we would be biased towards the annotator with the higher IAA score.

To assess the quality of our trained NER model, we computed the IAA between our authors' labels and the NER algorithm and the results were moderate, with an average of 0.66 amongst all the computations, showcasing an acceptable level of agreement between our authors' labels and the NER's extracted annotations.

Finally, we assessed the IAA computations between the gold labels and annotators. Annotators 2 and 3 had the highest agreements with the gold labels, which showed consistency in annotations. To evaluate the quality of the gold labels, we computed the IAA score between gold labels and authors' labels and the IAA score was 0.77, which showcased moderate quality. Our NER algorithm had similar IAA results when compared to gold labels.

It is important to note that, even if the IAA was low, it did not necessarily mean the agreements were low, since, in a sentence, the key entities and relations might appear several times and the annotators were free to choose which one to label.

4.2. Confusion Matrix

In addition to IAA scores, we also performed analysis on the difference in labelling between authors' labels and annotators in the form of confusion matrices. This is shown in Figures 4–6. Note that we only performed the confusion matrices analysis between the authors' labels and the annotators, and not with any other combinations in Table 5, because we treated our own labelling (authors' labels) as a form of ground truth. Therefore, by comparing the authors' labels and the annotators (via a confusion matrix), we were able to identify the type of "errors" the annotators were making.

Overall, the annotators performed well on Bidder, Target, Org, and terminated and the common confusion was of relational types considering, expecting, and success. Both annotator 1 and annotator 3 were great at identifying the considering relation and annotator 2 was good at identifying the expecting relation but seemed to be mistaken on the success and considering relations.

Although there was some confusion in labelling certain relations, the IAA results in Table 5, and the confusion matrices, show that the confusion was relative small and, as such, the annotations were reliable for us to train an accurate NER model to extract good entities and relations for our TKGQA dataset.

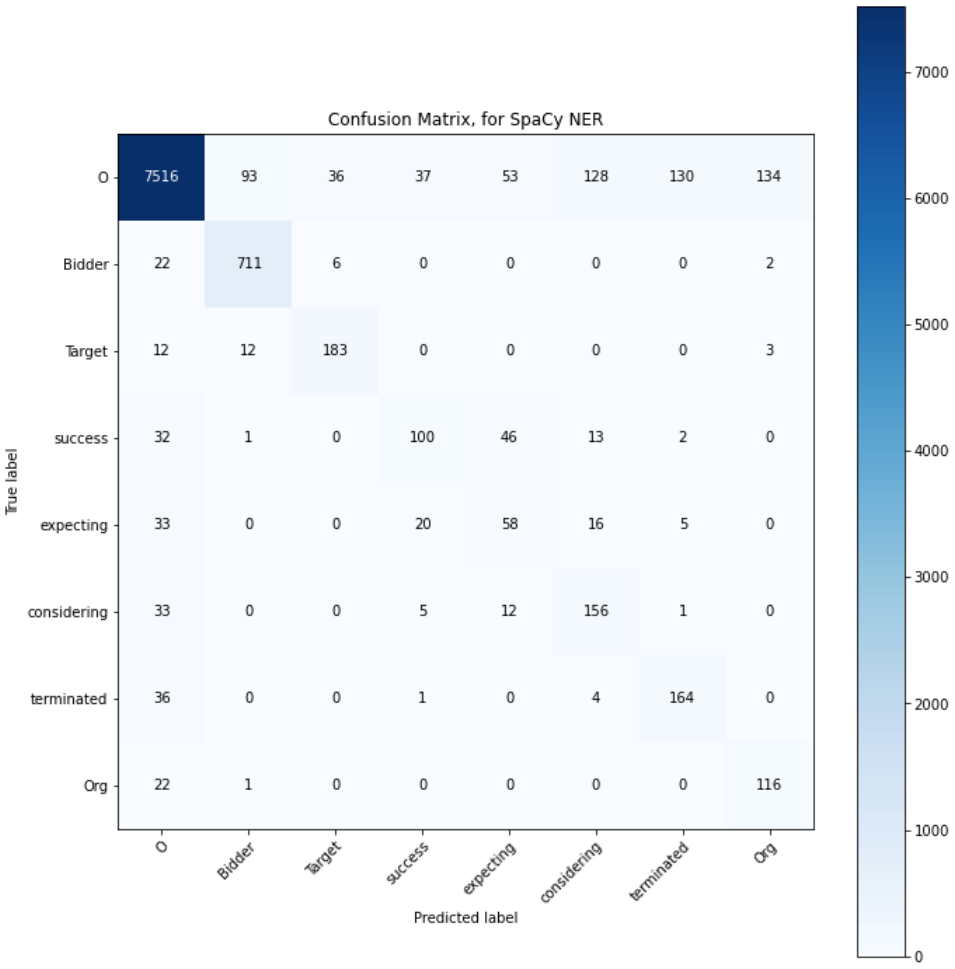

Figure 4. Confusion Matrix: Authors' Labels (True Label) vs. Annotator 1's Labels (Predicted Label).

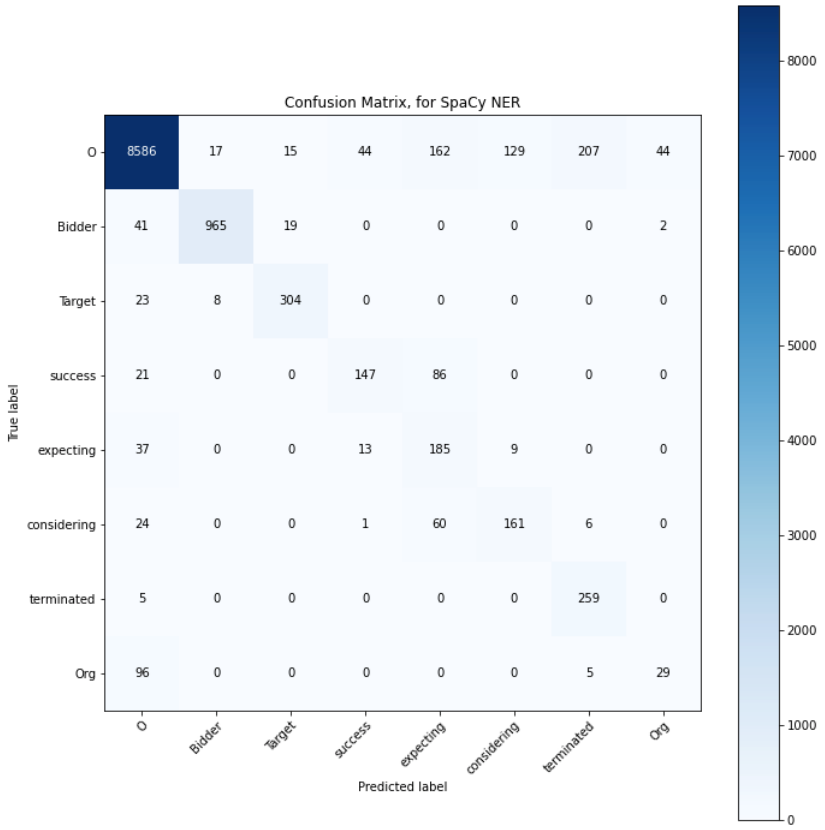

Figure 5. Confusion Matrix: Authors' Labels (True Label) vs. Annotator 2's Labels (Predicted Label).

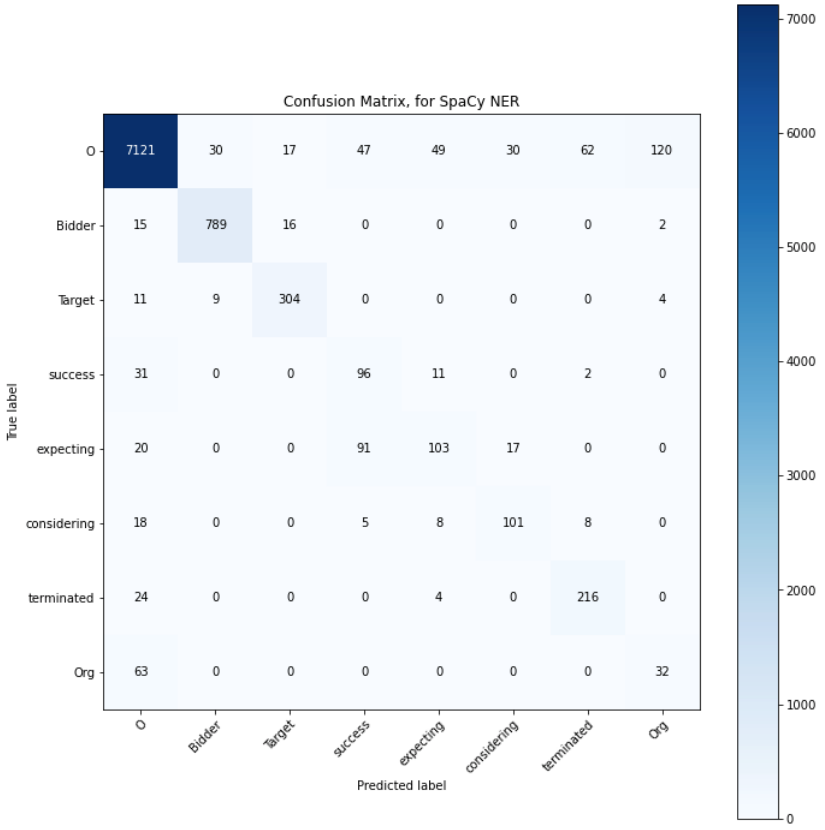

Figure 6. Confusion Matrix: Authors' Labels (True Label) vs. Annotator 3's Labels (Predicted Label).

5. User Notes

The primary use case of the TKGQA dataset is for the temporal knowledge graph binary classification update task; specifically, the use of question–answering to guide and validate the evolution of a temporal knowledge graph. With each document, we first update the TKG into a new TKG and then validate it using the questions and answers generated from the document. With the TKGQA dataset, we are better enabled in the important research area of validating the evolution of a temporal knowledge graph, ensuring that the knowledge graphs are being updated accurately with the latest information and, thus, that applications, such as recommendation systems, chatbots, semantic searches, etc., perform better and in a timely manner.

The primary use case requires the connection of four different components: (1) Temporal Knowledge Graph Embeddings, (2) Question Generation, (3) Temporal Knowledge Graph Update Task (Binary Classification), and (4) Question Answering over Temporal Knowledge Graph. In this way, the TKGQA dataset can also be used solely to research each of the individual components without any modifications. The modularity of our dataset allows researchers to easily experiment with different components. For example, much research in temporal knowledge graph completion involves transductive knowledge graph representation models [21–25], where the model has seen all the entities during training. However, this is unrealistic in real-world settings, as models are likely to encounter new entities and relations that they have not seen before. The TKGQA dataset accounts for this, and introduces new entities and relations in the validation and testing sets, to better facilitate research in developing inductive temporal knowledge graph embedding models to represent these zero-shot entities and relations.

Author Contributions: R.O. was responsible for conceptualising the overarching research objective, coming up with the methodology, leading data collection, annotation, and technical validation, and focusing on writing up and editing the final dataset paper. J.Ș. focused on acquiring financial support for the project in annotations and computing resources. O.S. and Y.-K.G. provided supervision on the overall project, giving feedback on research activities and writing. All authors have read and agreed to the published version of the manuscript

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: You can access the datasets through the Open Science Framework (OSF) repository at https://doi.org/10.17605/OSF.IO/XQWA4, accessible on 24 October 2022. We have included a README file to describe the details of each data record. Additionally, you can find the code that produces all the explanatory data analysis (EDA) work at https://github.com/RyanOngAI/m-a_temporal_knowledge_graph_qa, accessible on 24 October 2022.

Acknowledgments: This research was supported by the Royal Bank of Canada Wealth Management.

Conflicts of Interest: The authors declare no conflict of interests.

Abbreviations

The following abbreviations are used in this manuscript:

TKG	Temporal Knowledge Graph
M&A	Mergers and Acquisitions
Q&A	Questions and Answers
NER	Named Entity Recognition
EDA	Explanatory Data Analysis
IAA	Inter-Annotator Agreement
TKGQA	Temporal Knowledge Graph Question Answering

Appendix A

Table A1. TKGQA Example: Article ID 5.

Article ID 5	
Sentence Text	Under the terms of the transaction, upon completion of the acquisition, Bard became a wholly owned subsidiary of BD, and each outstanding share of Bard common stock was converted to the right to receive (1) $222.93 in cash without interest and (2) 0.5077 of a share of BD common stock.
Extracted Tuple	(‘bd_(company)’, ‘success_acq’, ‘bard’, ‘2018-01-02 00:00:00’)
General Questions	(“Who’s the bidder of the acquisition deal on 2018-01-02 00:00:00?”, ‘bd_(company)’, ‘entity’) (“Who’s the target of the acquisition deal on 2018-01-02 00:00:00?”, ‘bard’, ‘entity’) (“What’s the status of the deal between bd_(company) and bard on 2018-01-02 00:00:00?”, ‘success_acq’, ‘relation’)
After Head Entity Q&As	(‘Who does FlowJo LLC (Q106573956) belong to after the latest status of the deal between bd_(company) and bard?’, ‘bd_(company)’, ‘entity’) (‘Who owns Becton, Dickinson and Company headquarters (Q4878931) after the latest status of the deal between bd_(company) and bard?’, ‘bd_(company)’, ‘entity’)
After Tail Entity Q&As	[]
Before TKG	8321
After TKG	8322

Table A2. TKGQA Example: Article ID 124.

Article ID 124	
Sentence Text	Foresight has completed the acquisition of Canadian Solar’s Australian solar project pipeline.
Extracted Tuple	(‘foresight’, ‘success_acq’, ‘canadian solar’s australian solar project pipeline’, ‘2018-01-03 18:16:00’)
General Questions	(“Who’s the bidder of the acquisition deal on 2018-01-03 18:16:00?”, ‘foresight’, ‘entity’) (“Who’s the target of the acquisition deal on 2018-01-03 18:16:00?”, ‘canadian solar’s australian solar project pipeline’, ‘entity’) (“What’s the status of the deal between foresight and canadian solar’s australian solar project pipeline on 2018-01-03 18:16:00?”, ‘success_acq’, ‘relation’)
After Head Entity Q&As	[]
After Tail Entity Q&As	[]
Before TKG	8436
After TKG	8437

Table A3. TKGQA Example: Article ID 16719.

Article ID 16719	
Sentence Text	Walmart acquired a grocery wholesaler and distributor called McLane to manage its grocery distribution needs when Walmart first began to sell groceries in its stores.
Extracted Tuple	('walmart', 'success_acq', 'mclane', '2018-12-30 21:02:00')
General Questions	("Who's the bidder of the acquisition deal on 2018-12-30 21:02:00?", 'walmart', 'entity') ("Who's the target of the acquisition deal on 2018-12-30 21:02:00?", 'mclane', 'entity') ("What's the status of the deal between walmart and mclane on 2018-12-30 21:02:00?", 'success_acq', 'relation')
After Head Entity Q&As	('Who has influence over mclane after the latest status of the deal between walmart and mclane?', 'Marissa Mayer (Q14086)', 'entity') ('Who owns TodoDia (Q10382887) after the latest status of the deal between walmart and mclane?', 'walmart', 'entity') ("Who does Sam's Club (Q1972120) belong to after the latest status of the deal between walmart and mclane?", 'walmart', 'entity')
After Tail Entity Q&As	[]
Before TKG	12921
After TKG	12938

References

1. Ji, S.; Pan, S.; Cambria, E.; Marttinen, P.; Yu, P.S. A survey on knowledge graphs: Representation, acquisition and applications. *arXiv* **2020**, arXiv:2002.00388.
2. Trivedi, R.S.; Dai, H.; Wang, Y.; Song, L. Know-Evolve: Deep Temporal Reasoning for Dynamic Knowledge Graphs. In Proceedings of the International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017.
3. Cai, B.; Xiang, Y.; Gao, L.; Zhang, H.; Li, Y.; Li, J. Temporal Knowledge Graph Completion: A Survey. *arXiv* **2022**, arXiv:2201.08236.
4. Zelinka, M.; Yuan, X.; Côté, M.A.; Laroche, R.; Trischler, A. Building Dynamic Knowledge Graphs from Text-based Games. *arXiv* **2019**, arXiv:abs/1910.09532.
5. Das, R.; Munkhdalai, T.; Yuan, X.; Trischler, A.; McCallum, A. Building Dynamic Knowledge Graphs from Text using Machine Reading Comprehension. In Proceedings of the International Conference on Learning Representations, New Orleans, LA, USA, 6–9 May 2019.
6. Wijaya, D.; Nakashole, N.; Mitchell, T.M. CTPs: Contextual Temporal Profiles for Time Scoping Facts using State Change Detection. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 25–29 October 2014.
7. Tang, J.; Feng, Y.; Zhao, D. Learning to Update Knowledge Graphs by Reading News. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), Hong Kong, China, 3–7 November 2019.
8. Wang, A.; Cho, K.; Lewis, M. Asking and Answering Questions to Evaluate the Factual Consistency of Summaries. In Proceedings of the Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020.
9. Kazemi, A.; Li, Z.; Pérez-Rosas, V.; Mihalcea, R. Extractive and Abstractive Explanations for Fact-Checking and Evaluation of News. In Proceedings of the Fourth Workshop on NLP for Internet Freedom: Censorship, Disinformation, and Propaganda; Association for Computational Linguistics, Online, 6 June 2021; pp. 45–50. [CrossRef]
10. Arumae, K.; Liu, F. Guiding Extractive Summarization with Question-Answering Rewards. *arXiv* **2019**, arXiv:1904.02321.
11. Gunasekara, C.; Feigenblat, G.; Sznajder, B.; Aharonov, R.; Joshi, S. Using Question Answering Rewards to Improve Abstractive Summarization. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), Punta Cana, Dominican Republic, 7–11 November 2021.
12. Ning, Q.; Wu, H.; Han, R.; Peng, N.; Gardner, M.; Roth, D. TORQUE: A Reading Comprehension Dataset of Temporal Ordering Questions. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics: Online, 16–20 November 2020; pp. 1158–1172. [CrossRef]
13. Jia, Z.; Abujabal, A.; Saha Roy, R.; Strötgen, J.; Weikum, G. TempQuestions: A Benchmark for Temporal Question Answering. In Proceedings of the Companion Proceedings of The Web Conference, Lyon, France, 23–27 April 2018; pp. 1057–1062. [CrossRef]

14. Souza Costa, T.; Gottschalk, S.; Demidova, E. Event-QA: A Dataset for Event-Centric Question Answering over Knowledge Graphs. In Proceedings of the 29th ACM International Conference on Information & Knowledge Management, Association for Computing Machinery: New York, NY, USA, 19–23 October 2020; p. 3157–3164. [CrossRef]
15. Jia, Z.; Abujabal, A.; Roy, R.S.; Strotgen, J.; Weikum, G. TEQUILA: Temporal Question Answering over Knowledge Bases. In Proceedings of the 27th ACM International Conference on Information and Knowledge Management, Turin, Italy, 22–26 October 2018.
16. Wu, W.; Zhu, Z.; Lu, Q.; Zhang, D.; Guo, Q. Introducing External Knowledge to Answer Questions with Implicit Temporal Constraints over Knowledge Base. *Future Internet* **2020**, *12*, 45. [CrossRef]
17. Saxena, A.; Chakrabarti, S.; Talukdar, P.P. 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, Bangkok, Thailand, 1–6 August 2021.
18. Ong, R.; Sun, J.; Serban, O.; Guo, Y.K. TKGQA Dataset: https://doi.org/10.17605/OSF.IO/XQWA4 (accessed on 24 August 2022)
19. Nakayama, H.; Kubo, T.; Kamura, J.; Taniguchi, Y.; Liang, X. doccano: Text Annotation Tool for Human. 2018. Software. Available online: https://github.com/doccano/doccano (accessed on 24 October 2022).
20. Cohen, J. A Coefficient of Agreement for Nominal Scales. *Educ. Psychol. Meas.* **1960**, *20*, 37–46. [CrossRef]
21. García-Durán, A.; Dumančić, S.; Niepert, M. Learning Sequence Encoders for Temporal Knowledge Graph Completion. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics: Brussels, Belgium, 31 October–4 November 2018; pp. 4816–4821. [CrossRef]
22. Goel, R.; Kazemi, S.M.; Brubaker, M.; Poupart, P. Diachronic Embedding for Temporal Knowledge Graph Completion. In Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence, New York, USA, 7–12 November 2020.
23. Lacroix, T.; Obozinski, G.; Usunier, N. Tensor Decompositions for temporal knowledge base completion. In Proceedings of the Eighth International Conference on Learning Representations: Online, 26 April–1 May 2020
24. Messner, J.; Abboud, R.; Ceylan, I.I. Temporal Knowledge Graph Completion using Box Embeddings. In Proceedings of the AAAI Conference on Artificial Intelligence, Online, 22 February–1 March 2021.
25. Xu, C.; Nayyeri, M.; Alkhoury, F.; Shariat Yazdi, H.; Lehmann, J. TeRo: A Time-aware Knowledge Graph Embedding via Temporal Rotation. In Proceedings of the 28th International Conference on Computational Linguistics, International Committee on Computational Linguistics: Barcelona, Spain (Online), 13–18 September 2020; pp. 1583–1593. [CrossRef]