A Fundamental Evaluation of Candidate Answers Generation for Question Answering Using Wikidata

Ryoga Nakagawa[1,*,†], Kouji Kozaki[1,*,†]

[1]Osaka Electro-Communication University, 18-8 Hatsuchou, Neyagawa-shi, Osaka, Japan, 572-0833

Abstract

Question answering (QA) is an import application of knowledge graphs (KGs). In order to evaluate knowledge structures within a knowledge graph, we consider an investigation into the details of correspondences between questions and knowledge graph. In this paper, we focus on Wikidata as a knowledge graph in the general domain and evaluate a basic question answering method using Wikidata, aiming to conduct a fundamental evaluation of its content as a knowledge graph.

Keywords

Knowledge graph, Wikidata, Question answering, Natural language question answering, SPARQL

1. Introduction

Knowledge graphs are widely used in various fields and have been applied to tasks such as natural language processing, question answering, information retrieval, and problem-solving. In question answering (QA), a knowledge graph enables users to ask questions in natural language, and the system generates answers based on the knowledge graph. Specifically, the system interprets the meaning of a question, extracts relevant entities and relationships from the knowledge graph, and combines them to generate answers.

Several existing works focus on developing query answering systems using knowledge graphs. For instance, T. Ploumis et al. developed a query answering system using Wikidata based on the dependency analysis of question text[1]. X. Hu, et.al proposed a method for question answering over knowledge graph based on SPARQL queries and keyword search [2]. Such an approach plays an important role in improving the performance of QA systems. These studies utilize datasets from various domains for QA and conduct experiments to evaluate the proposed systems.

However, to evaluate knowledge structures within a knowledge graph, we are considering a more in-depth investigation into correspondences between questions and knowledge graph. In this paper, we uses Wikidata [3] as an open knowledge graph in the general domain and evaluate a basic question answering method using Wikidata, aiming to conduct fundamental evaluation of its contents as a knowledge graph.

IJCKG 2023 Poster and Demo track

[*]Corresponding author.

[†]These authors contributed equally.

mi23a002@oecu.jp (R. Nakagawa); kozaki@osakac.ac.jp (K. Kozaki)

0000-0003-4578-4980 (K. Kozaki)

CEUR Workshop Proceedings (CEUR-WS.org)

2. Test Dataset for Question Answering using Wikidata

In this study, we have a test dataset for question answering using Wikidata. While there are several datasets for question answering using Wikidata, they do not support Japanese language. Therefore, we utilized the Japanese quiz dataset[4] which was developed using Japanese Wikipedia for a competition of QA system primarily focused on NLP. This dataset provides provides question text and its answer, which is described as one of Wikipedia articles. Consequently, we can also obtain Wikidata ID corresponding to the article because Wikidata contains resources related to all Wikipedia articles in principle. In order to determine how many questions can be correctly answered using Wikidata, we conducted manual SPARQL queries. It's important to note that some questions cannot be answered using any SPARQL query due to the limitations in the graph structures of Wikidata, which may lack the necessary knowledge to answer certain questions. From the quiz dataset, We choose 50 questions and formulated SPARQL queries. As a result, we were able to create SPARQL queries that provided correct answers for 33 out of the 50 questions.

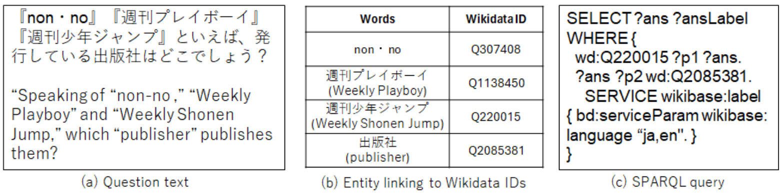
3. Evaluation of Candidate Answers Generation

To generate candidate answers, it is necessary to map the words in the question text to Wikidata IDs and then use the mapping results to generate SPARQL queries to find answers using Wikidata. The former corresponds to Entity Linking, and the latter to SPARQL query generation. In the following sections, we evaluate these two methods using the test dataset provided in Section 2. As an example, we will use the question shown in (Figure 1 (a)) to illustrate these methods.

3.1. Evaluation of Entity Linking

Out of 50 question in the test dataset, we evaluated entity linking using the 33 questions for which we could obtain correct answers using SPARQL queries on Wikidata. For the entity linking method, we extracted words from question text and searched Wikidata for entities that matched the extracted words using an exact string match. In the example question mentioned above, four entities with Wikidata IDs were obtained from the search results (see Figure 1 (b)). The evaluation was conducted manually using Wikidata search API to avoid any potential issues caused by morphological analysis.

Next, we compare the obtained IDs from a question text to the SPARQL query for the question in the test dataset. If all the IDs in the query are included in the obtained IDs, it means that the



『non・no』『週刊プレイボーイ』『週刊少年ジャンプ』といえば、発行している出版社はどこでしょう？

"Speaking of "non-no ," "Weekly Playboy" and "Weekly Shonen Jump," which "publisher" publishes them?

(a) Question text

Words	Wikidata ID
non・no	Q307408
週刊プレイボーイ (Weekly Playboy)	Q1138450
週刊少年ジャンプ (Weekly Shonen Jump)	Q220015
出版社 (publisher)	Q2085381

(b) Entity linking to Wikidata IDs

```
SELECT ?ans ?ansLabel
WHERE {
 wd:Q220015 ?p1 ?ans.
 ?ans ?p2 wd:Q2085381.
   SERVICE wikibase:label
{ bd:serviceParam wikibase:
language "ja,en". }
}
```

(c) SPARQL query

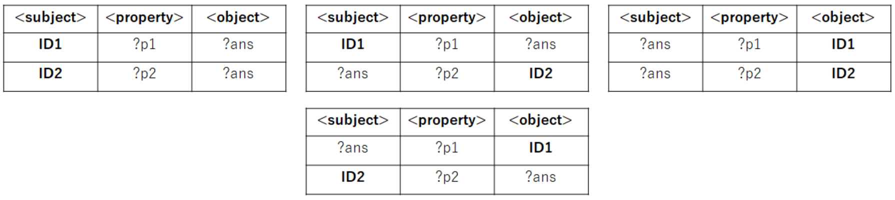
Figure 1: An example of (a)question text, (b)entity linking and (c)SPARQL query

query could be generated using them. In the case of the example of Figure 1(a), all the IDs shown in the query in Figure 1(c) are included in the obtained IDs shown in in Figure 1(b). However, it's worth noting that Wikidata IDs of properties are not obtained by the entity linking method, so they are replaced with variables such as "?p1","?p2" in the query. In the evaluation results of entity linking, it was found that for 27 of the 33 questions, all the IDs required to generate the SPARQL queries were obtained. However, for 6 questions, the questions text included some IDs that were not obtained by the entity linking method due to differences in spelling between the question text and Wikidata.

3.2. Evaluation of SPARQL Query Generation

For the 27 questions which all the required IDs could be obtained through entity linking, we evaluate SPARQL query generation to obtain the candidate answers using the graph structure of Wikidata. In this study, we explore an exhaustive combination of the entities obtained through entity linking. When two IDs are obtained through entity linking, four SPARQL patterns are generated. For instance, when two IDs are obtained, such as ID1 and ID2, four patterns of SPARQL queries are generated as illustrated in Figure 2. In this exampel, variables like "?p1","?p2" represent properties, and "?ans" represents the answer to the question. If the SPARQL query accurately represents the question's condition, the correct answer can be returned as the value of "?ans" . However, if there are multiple entities that satisfy the search conditions, a single query pattern can yield multiple candidate answers.

To evaluate the query generation, we executed the generated queries on SPARQL endpoint of Wikidata and verified whether their query results contained the correct answer to the questions. As a results, out of the 27 questions, the method successfully generated SPARQL queries that produced the correct answer for 23 questions. However, in the case of the remaining four questions, timeouts occurred during SPARQL query processing, preventing us from obtaining the correct answers..



<subject>	<property>	<object>
ID1	?p1	?ans
ID2	?p2	?ans

<subject>	<property>	<object>
ID1	?p1	?ans
?ans	?p2	**ID2**

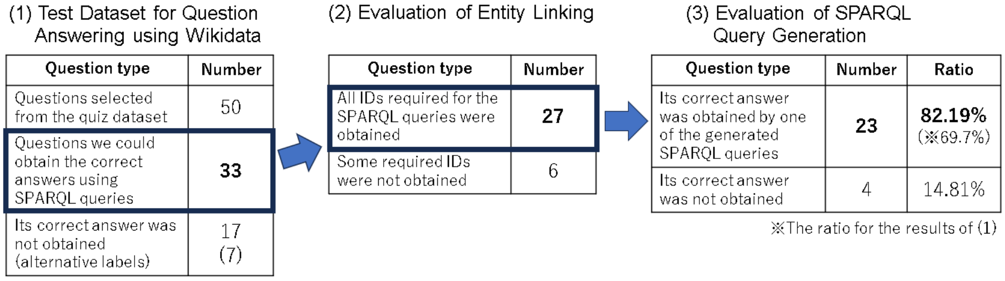
<subject>	<property>	<object>
?ans	?p1	**ID1**
?ans	?p2	**ID2**

<subject>	<property>	<object>
?ans	?p1	**ID1**
ID2	?p2	?ans

Figure 2: Patterns for SPARQL query generation using two IDs

4. Discussion

Figure 3 displays an overview of the evaluation results. For our testing, we used 50 questions from the Japanese quiz dataset. Through manual efforts, We obtained SPARQL queries capable of deriving correct answers from Wikidata for 33 of these questions (Figure 3(1)). These 33 questions with SPARQL queries constitute the correct data for evaluating question answering using Wikidata. Among the remaining 17 questions, 7 required aliases (alternative labels) to



(1) Test Dataset for Question Answering using Wikidata

Question type	Number
Questions selected from the quiz dataset	50
Questions we could obtain the correct answers using SPARQL queries	**33**
Its correct answer was not obtained (alternative labels)	17 (7)

(2) Evaluation of Entity Linking

Question type	Number
All IDs required for the SPARQL queries were obtained	**27**
Some required IDs were not obtained	6

(3) Evaluation of SPARQL Query Generation

Question type	Number	Ratio
Its correct answer was obtained by one of the generated SPARQL queries	**23**	**82.19%** (※69.7%)
Its correct answer was not obtained	4	14.81%

※The ratio for the results of (1)

Figure 3: The overview of evaluation results

obtain the correct answers using SPARQL queries to Wikidata. For the remaining 10 questions, we could obtain the correct answers using SPARQL due to the lack of knowledge defined in Wikidata. We evaluated fundamental methods of entity linking and SPARQL query generation using the 33 questions and their associated queries in the correct dataset.

Through the evaluation of entity linking, we obtained all the IDs required for the SPARQL queries for 27 of the 33 questions. We then applied these IDs to query generation and obtained the correct answers using one of the generated SPARQL queries for 23 of the 27 questions. In other words, we were able to obtain the correct answers from Wikidata for 82.2% of the questions when we could obtain enough IDs through entity linking. We believe that we could obtain correct answers for the remaining questions by improving the querying performance because the primary reason why we could not obtain the answers was the timeout problem during SPARQL query processing. When we compare the final results in (3) and the 33 questions in the correct dataset, we found that we could obtain correct answers from Wikidata for 69.7% of questions. To improve these results, we need to enhance the accuracy of entity linking.

In this way, this study conducts a fundamental evaluation of candidate answers generation for question answering using Wikidata. The evaluation results could serve as baseline data to development of question answering methods using Wikidata. As part of our future works, we plan to extend the number of questions for the test datasets and evaluation results. We will consider using not only manual efforts but also automatic methods to prepare them using entity linking and SPARQL query generation, as discussed in Section 3. Additionally, we are in the process of developing a QA system by extending these methods.

References

[1] T. Ploumis, I. Perikos, F. Grivokostopoulou, I. Hatzilygeroudis, A factoid based question answering system based on dependency analysis and wikidata (2021) 1–7.

[2] X. Hu, J. Duan, D. Dang, Natural language question answering over knowledge graph: the marriage of sparql query and keyword search, Knowledge and Information Systems 63 (2021) 819–844.

[3] Wikidata, https://www.wikidata.org/, 2022. Accessed on 2022/10/25.

[4] Jaqket japanese on the subject of quizzes qa data-set, https://jaqket.s3.ap-northeast-1.amazonaws.com/data/aio_01/dev1_questions.json, 2020. Accessed on 2022/10/25.