2O5-E-3-02

An Approach to Knowledge Graph Completion based on Discussion Agents using IBIS Structure

Xiangyu Zhang[*1] Shun Shiramatsu[*2]

[*1] Department of Computer Science, Faculty of Engineering, Nagoya Institute of Technology

[*2] Department of Computer Science, Graduate School of Engineering, Nagoya Institute of Technology

Knowledge Graph Completion Challenge 2018, a competition of interpretable AI systems to solve crime story, was held and we participated in it. Our approach is based on discussion agents using IBIS (issue-based information system), a kind of discussion structures. This paper is a part achievement of the design architecture. We design two types of agents: a discussion agent and a facilitation agent. The discussion agents generate hypotheses about the criminate. The facilitator agent ask questions to clarify the detail of the hypotheses. To manage the discussion on the hypotheses, IBIS structure is suitable because it has better interpretability. This approach based on the hypothesis generation has a possibility to be also utilized in the real-world discussion support.

1. Introduction

In recent years, along with the spread of deep learning, which process is difficult to understand, it is expected that the reason of deep learning result can be explainable. The need of artificial intelligence (AI) with interpretability is increasing.

Under such a background, Knowledge Graph Completion Challenge (KGC), a contest to recruit ideas and solutions, was held in November 2018. In this contest, Sherlock Holmes' mystery novel "Speckled Band" was taken as a theme, and a knowledge graph [kgc 18] describing its contents in RDF triple (KGC Data) was provided. The purpose of this completion is not only to find the correct answer through reasoning, but also more importantly, to give a compelling reason when interpreting the results. This means that it looks forward to establishing explainable and interpretable artificial intelligence.

A paper in last November [Shiramatsu 18] provided a design architecture, which is expected to give an idea to achieve the goal of KGC. The focus of this paper is to implement it this time. Here are some technical issues to solve:

- How to develop the generation logic of a hypothesis?
- What questions do the agents ask about a hypothesis?
- How can the facilitator agent evaluate the consistency of the hypothesis?

2. Related Work

2.1 Architecture for Knowledge Graph Complement

The design architecture gives a guide step by step to make sure to complete a system to reason murderer and motivation out from a mystery novel, the system is based on discussion agents.

Common knowledge is needed in this architecture since KGC Data only includes information of novel. And agents that can give multiple hypotheses are necessary.

When discussing, agents' actions include giving hypotheses, asking questions about a hypothesis, investigating clues, giving evidence.

It is also excepted that an agent can challenge other agents' hypotheses and give defenses.

After all discussion over, there will be a facilitator agent who evaluates the various hypotheses proposed by the previous agents, and then gives the score. The hypothesis with the highest score is regarded as the truth, and the suspect in the hypothesis is regarded as the prisoner.

2.2 IBIS Structure

Issue-based Information System (IBIS) structure [Kunz 70] offers a visual way to check agents' movement and system processing. Every time an agent has an action, there will be a node to represent it in IBIS structure. We use Web API of IBIS CREATOR, which is a Web-based IBIS editor implemented by Kamiya [Kamiya 19], for structuring discussion among the agents..

3. Material and Approach

3.1 Tools and Environment

Virtuoso is a server that combines relational, graph, and document data management with web application server and web services platform functionality. In this project, agents have to keep common knowledge besides KGC Data.

According to the instructions of the competition, the execution of reasoning should be carried out under three conditions to test the effect of reasoning in different situations:

a. using full knowledge graphs

b. using data with id numbers below 368 (10% incomplete)

c. using data with id numbers below 268 (25% incomplete)

It's hard to separate three execution conditions in one RDF triple data file. So the data file is downloaded, copied and deleted according to the execution conditions. The virtuoso server becomes necessary to give them different Graph IRI values to as a distinction.

In this system, Virtuoso is also used when agents use common knowledge for reasoning. The settings of Graph IRI this time is as Table 1.

Contact: Xiangyu Zhang, Nagoya Institute of Technology, Gokiso-cho, Showa-ku, Nagoya-shi 466-8555, E-mail: zhang.xiangyu@srmtlab.org

Table 1: Graph IRI

Data	Graph IRI
full knowledge graph	http://kgcdata/all
data with id below 368	http://kgcdata/368
data with id below 268	http://kgcdata/268
common knowledge	http://commonknowledge

For connecting KGC Data, this time program uses SPARQLWrapper library. SPARQLWrapper is a simple Python wrapper around a SPARQL service to remotely execute queries. It can help in creating the query invocation and, possibly, convert the result into a more manageable format.

3.2 Common Knowledge

It is necessary to add common knowledge. Since knowledge graph only include necessary information of novel, adding common knowledge can help agents understand things more like the real world.

There is a sample of common knowledge:

In the definitions of subjects "sister", "Julia" and "Helen", there are only two attributes for each other: "rdf:type" and "rdfs:labels". But if checking KGC Data, it is easy to know subject "sister" includes "Julia" and "Helen". So the relationship information among them is added. Table 2 shows this repair.

3.3 Program Design

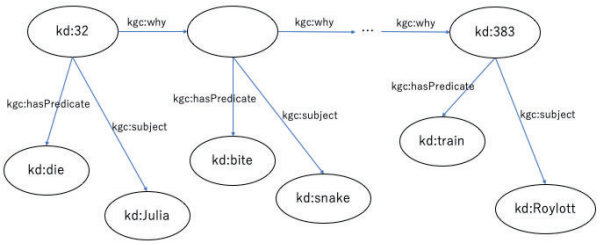
Three technical issues were raised in the introduction section, here the first two are answered.

The first is, how to develop the generation logic of a hypothesis?

The agents will give a variety of hypotheses. There must be a logical chain to explain why a hypothesis is generated. The chain usually includes many rings. Here, agents use a chain called "why-chain". The chain represents the generation logic of a hypothesis. A classic scene is, before generating a hypothesis, the agents will look for the cause of death of a victim, there will be multi-steps to conclude the cause.

There is an explanation of “why-chain”.

For example, to speculate a victim's death, first of all, it must clarify what will be the cause of death. It includes a fatal injury, being poisoned, frozen or scared to death. The second step, the symptoms of each cause are different. If it is a fatal injury, the body of the victim must have a wound. If it is being poisoned, the victim will behave symptoms such as dizziness or vomiting before dying. If it is frozen to death, the skin of the victim may



appear red. Next step, if the symptoms by the victim are likely to be poisoned, it may be one of these situations: poison, poisonous plants or bitten by poisonous animals [Matsushita 18]. Connect each ring in the chain with “kgc:why” and form a “why-chain”. Figure 1 is the "why-chain" of the hypothesis “How did Julia die”.

Make a summary of what may happen in the real world and put it in common knowledge. Before the agents give a hypothesis, they scan the KGC Data and common knowledge to make a matching. Then agents will give a hypothesis. The same technique can also be used to speculate a motivation of a suspect.

The second issue was raised in the introduction section is, what questions do the agents ask about a hypothesis?

Widely known, 5W1H are questions whose answers are considered basic in problem solving. They are often mentioned in police infestations. Their advantage is that none of them can be answered with a simple "yes" or "no". When agents do a reasoning, they use one or more to ask further questions.

The system includes several types of agents: the basic agent is named with BasicAgent, it has some common properties of all kinds of agents, such as "agent_id" "agent_name", "agent_type" and so on.

Then there are two main agents named DiscussionAgent and FacilitatorAgent. DiscussionAgent generates a hypothesis about who is the criminal. FacilitatorAgent asks questions according to 5W1H on the hypotheses, and DiscussionAgent gives sub-hypotheses as answers. There not only one instance of DiscussionAgents. They will be distinguished by id or name property.

There is a flag that is used to mark whether a hypothesis and its sub-hypotheses have all been discussed and answered. If it isn’t finished yet, the value of mark is "1", and the discussion

Table 2: repair of KGC Data

Resource	KGC Data	Repair
kd:sister	rdf:type kgc:person; rdfs:label “sister”@en;	rdf:type kgc:person; rdfs:label “sister”@en; kgc:ofPart <http://kgc.knowledge-graph.jp/data/SpeckledBand/Julia>; kgc:ofPart <http://kgc.knowledge-graph.jp/data/SpeckledBand/Helen>;
kd:Julia	rdf:type kgc:person; rdfs:label “Julia”@en;	rdf:type kgc:person; rdfs:label “Julia”@en; kgc:ofWhole <http://kgc.knowledge-graph.jp/data/SpeckledBand/sister>;
kd:Helen	rdf:type kgc:person; rdfs:label “Helen”@en;	rdf:type kgc:person; rdfs:label “Helen”@en; kgc:ofWhole <http://kgc.knowledge-graph.jp/data/SpeckledBand/sister>;

kd is a prefix for representing <http://kgc.knowledge-graph.jp/data/SpeckledBand/>

Table 3: remove subjects to narrow down suspect

Subject	Reason to remove
man	In KGC Data, the meaning of “man” is someone, it’s not the name of a specific person. Similarly, the “suspect” here is also a generic suspect. It just was used when the detective talks with his assistant or other characters in novel, not referring to someone who exists.
suspect	
Holmes	The premise of this reasoning process is that the detective and his assistant will not become murderers, so they are removed here. Of course, this situation is not absolutely true in reality, this issue will be discussed again in part of future work.
Watson	
mother-of-sister	The reason of removing them is that there are synonyms for these three subjects in KGC Data. That is, “mother-of-sister” is equal to “mother-of-Helen”, “father-in-law” is equal to “Roylott”, “friend-of-Roylott” is equal to “Roma”. In the process of reasoning, the words “mother-of-Helen”, “Roylott” and “Roma” appear far more frequently than the previous three synonyms. Using these three words will also make reasoning easier.
father-in-law	
friend-of-Roylott	

should be continued. If all doubtful points of the hypothesis are already discussed, then the value of mark turns to "0", meanwhile the discussion is completed, it should be summarized to report to the father hypothesis or gives a conclusion.

Finally, FacilitatorAgent evaluates the hypothesis.

3.4 Process

It is easy to extract all victims and characters from KGC Data using the SPARQL statement. The agent's action is to select one of victims to ask questions. The initial question is definitely "Who killed this dead person". To answer this question, according to the design structure, another agent should be expected to do a direct product using victims and all characters in the novel. But there is one point: in KGC Data, several of subjects which value of "kgc:type" is "person" are not needed.

Totally they are seven: "man", "suspect", "Holmes", "Watson", "mother-of-sister", "father-in-law", "friend-of-Roylott". Here Table 3 tells why these seven subjects are removed.

The elimination for these seven subjects will be done by agents using common knowledge actually. now the left subjects are called suspects. Agent will make a direct product between the victim who has been raised as a topic with the suspects. The form is isKiiledBy(victim, suspect) and it is passed on to other agents.

In processing, DiscussionAgent moves first, it gives a hypothesis once. With original hypothesis, FacilitatorAgent begins to ask questions according to 5W1H, they are "what" of 5W1H, "What could the suspect gain after killing the victim?", and "how", "How did the suspect kill the victims?". Then DiscussionAgent give sub-hypotheses for questions.

The why-chain is prepared for generating hypothesis for “how”, and there is also an idea to generate a hypothesis for “what”. Figure 2 is a view to explain the logic of how to generate a hypothesis of “What could Roylott gain after killing Julia”.

Firstly, agents catch all situations where “kgc:subject” is equal to “Julia” or “Roylott” and assign it to set A. Meanwhile agents catch situations that include “Julia” and “Roylott” at the same time and assign it to set B. Secondly, agents match special situations from set A or B using a special word list. If agents match succeed, they will give a hypothesis, if failed, agents will say “Roylott’s motivation is not clear”.

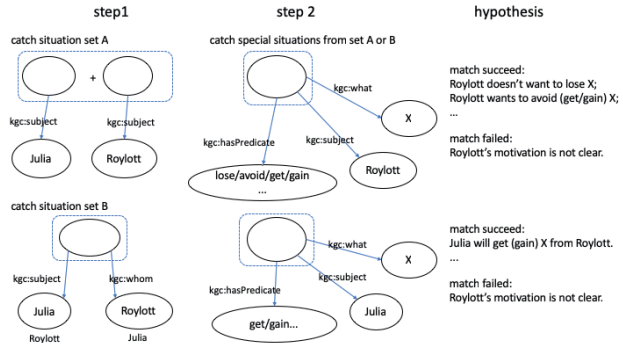


Figure 2: generate a hypothesis for question “what”

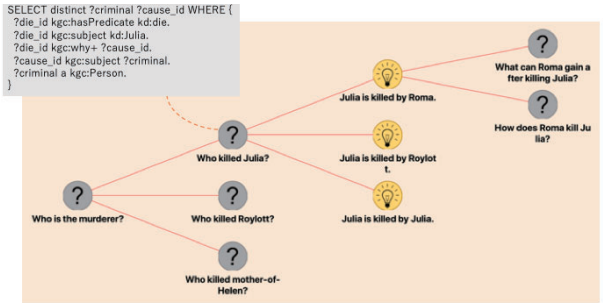


Figure 3: IBIS Structure

4. Conclusion and Future Work

In the Introduction section, three issues are excepted to be solved, then the key ideas of the previous two issues are provided in the Program Design section. To deal with how to develop the generation logic of a hypothesis, the key idea is to use a chain called "why-chain", a chain represents the generation logic of a hypothesis. To deal with what questions do the agents ask about a hypothesis, the key idea is that FacilitatorAgent will ask questions according to 5W1H.

In the current version of the implementation, every time an agent gives a hypothesis randomly. Other agents will ask sub-questions about it. Figure 3 is a view of agents’ actions with IBIS Structure.

Up to now, this paper's work is a half-finished product of the designed architecture. We will continue the development to enable agents to generate consistent hypotheses and answers for all questions.

And it is expected agents which have different hypotheses can challenge each other. It means they can give opposing views to others, this will make it a real discussion.

It is known that discussion among agents is not proposed to convince each other but to speaking to the third side. Here the third side is the facilitator agent. So how can the facilitator agent evaluate the consistency of the hypothesis? This is an important task for this system. This technical issue was generated in the introduction section but not answered yet, actually it is still under consideration.

If the methodology to generate sub-hypotheses to answer questions is established in the future, it has a possibility to be applied to supporting real-world discussion because such method can be used for giving logic behind discussion participants' ideas. Since our research team is currently developing a facilitator agent for supporting public debate [Kitagawa 19, Ikeda 17], we will also consider such direction towards the discussion support technology.

Acknowledgements

This work was partially supported by JSPS KAKENHI (17K00461).

References

[kgc 18] Knowledge Graph of "SpeckledBand", https://github.com/KnowledgeGraphJapan/Challenge/tree/master/rdf/SpeckledBand, 2018 (in Japanese).

[Shiramatsu 18] S. Shiramatsu , X. Zhang , A. Kamiya , M. Watanabe: Designing an Architecture for Knowledge Graph Completion based on Discussion Agents using IBIS Structure, in *Proc. of SIG-SWO*, 46(2), SIG-SWO-046-02, 2018. (in Japanese)

[Kunz 70] W. Kunz and HWJ. Rittel: Issues as Elements of Information Systems. in *Berkeley, California: Institute of Urban and Regional Development, University of California*, CiteSeerX 10.1.1.134.1741, 1970.

[Kamiya 19] A. Kamiya, K. Kitagawa, S. Shiramatsu, D. Shibata, K. Yoshino, S. Suzuki, T. Ito: Prototypes for Visualizing IBIS Structure and Discussion Progress towards Supporting Consensus Building in Web-based Discussion, in *Proc. of IPSJ 2019*, 1ZB-08 (in Japanese)

[Matsushita 18] Kamikotanaka 411 (K. Matsushita, T. Kaneko, K. Yoshikawa, K. Kobayashi, Y. Koyanagi, T. Ukai, F. Nishino, M. Oda): Knowledge Graph Completion Challenge, http://challenge.knowledge-graph.jp/results/results2018.html#2nd, 2018 (in Japanese)

[Kitagawa 19] Ko Kitagawa, Shun Shiramatsu, Akira Kamiya: Developing a Method for Quantifying Degree of Discussion Progress towards Automatic Facilitation of Web-based Discussion, in *Multi-Agent Systems and Agreement Technologies 16th European Conference, EUMAS 2018, and 6th International Conference, AT 2018, Revised Selected Papers*, Springer LNCS, 2019 (to appear)

[Ikeda 17] Yuto Ikeda, Shun Shiramatsu: Generating Questions Asked by Facilitator Agents Using Preceding Context in Web-based Discussion, in *Proceedings of the 2nd IEEE International conference on Agents*, pp. 127-132, 2017.