Action Planning based on Open Knowledge Graphs and LOD

Seiji Koide[1], Fumihiro Kato[1], Hideaki Takeda[1,2], Yuta Ochiai[3], and Kenki Ueda[3]

[1] National Institute of Informatics, 2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo 101-8430, Japan,
takeda@nii.ac.jp,
WWW home page: http://www-kasm.nii.ac.jp/~takeda/index.html
[2] SOKENDAI (The Graduate University for Advanced Studies)
[3] Toyota Motor Corporation

Abstract. In this preliminary report, we show how we can realize action planning by using open knowledge-bases and LOD like Linked Geo Data, DBpedia, and WordNet, etc. To make a recommendation for car drivers and passengers, we combine these open datasets by newly constructed ontologies of facilities and services. Then we develop the inference procedure to translate user requests into SPARQL queries to obtain a recommendation on appropriate facilities and areas for users. Common sense knowledge is also required in the reason process.

Keywords: DBpedia, LinkedGeoData, Knowledge-based system

1 Introduction

While Linked Data is now gradually growing to be the infrastructure of coming Knowledge Society, we are still struggling to show the potential of Linked Data to most people in basic industries. To cope with this situation and propel the deployment of Semantic Web technology in the society, it is needed to demonstrate the performance of linking distinct datasets and show the potential and usefulness of outbound and inbound linking data beyond enterprise data in higher levels of diverse applications. However, although each collection of large linked data such as DBpedia, Freebase, and OpenCyc are a kind of isolated showcase of LOD with internally linked data within their own territory and objective, yet there is no linking data among them from the viewpoint of LOD applications.

In this preliminary work, we utilized linked open datasets, DBpedia, Linked Geo Data, and WordNet for the purpose of making a recommendation system for car drivers and passengers. We have found that it is required more goal-oriented linked datasets and common sense knowledge as bridge between isolated LOD datasets. We have also found that Semantic Web technology or specifically LOD and SPARQL engines are enough as enabling technology to create and demonstrate new applications based on heterogeneous and diverse datasets.

In our use-case, the system accepts ambiguous requests from car drivers and passengers, plans driver actions to achieve goals that satisfies the requests, including alternatives, and makes a recommendation for the drivers and passengers.

To obtain the destination as goal, we utilized Linked Geo Data and DBpedia, and arranged them with newly constructed facility ontology and service ontology for linking among such open datasets. WordNet is also utilized as general knowledge, because it was necessary to make the inference with common sense to discover driving destinations from user requests. Then, we developed the inference procedure to translate user requests into SPARQL queries to obtain a recommendation on appropriate facilities and areas for users.

The purpose of this preliminary report is to make a clear direction for development of LOD applications in order to deploy linked data as the infrastructure of society in future.

2 Problem Setting for the Use Case

In setting of the use-case, we firstly made more than ten scenarios of conversation between users and this system. In each case, a user in a car speaks a single or a number of requests to do something with driving a car. Then, the system analyzes the requests under the consideration of current contexts such as time, location, driving time, etc. At last, the system makes concrete action proposals to visit specific points (shop, facility, etc.) or areas (sightseeing area, good place for time-consuming, etc.) with a reasonable visiting order. Basically, the request may be vague and complex, but the recommendation is specific and concrete. However, every recommendation is a sequence of actions, and proposed actions are quite limited within these scenarios, for example, drive somewhere, buy or eat something, do some sport, and so on. One of the simplest scenarios is as follows.

> Child passenger(hereafter C): I want to see a lion.
> System(hereafter S): How about Ueno Zoo. A baby lion was born recently.
> C: It sounds good, but I was there last month.
> S: Well, how about Kinoshita Circus. You can see a lion show there.
> C: OK. That's fine.

In this scenario, the system must discover the knowledge that a lion is a kind of animal and a zoo is an public entertainment facility for seeing animals. The system must find out a nearest zoo, that is Ueno Zoo in this case, from the current location, and must reason that users have enough time to drive to the destination and walking around the zoo. Furthermore, due to the negative response of the user, the system must discover a neighboring circus that presents a lion show as an alternative.

3 Ontologies for Facility, Action Target, and Service

Instead of directly searching individual facilities like Ueno Zoo or individual shops like Yodobashi Akiba store (a home electric appliance mass retailer in Japan), we considered classes of facilities like zoo or home electric appliance mass retailer to make the system scalable, then made a facility ontology that contains typical facilities and we defined typical users' behavior at such facilities like "a user sees animals in a zoo" or "a user buys a household appliance at a home electric appliance mass retailer". Even if we accidentally fail to guide an actual facility that satisfies user's special requests, such a problem will be solved with the development of more rich and specific datasets that includes individual facilities.

The facility ontology is constructed mainly by extracting facility classes related to leisure and meals in Lined Geo Data (LGD). LGD constructs a shallow class hierarchy from tags attached to the nodes and ways of OpenStreetMap (OSM). Therefore, LGD classes makes it easy to incorporate new facilities and new facility types.

On the other hand, as a result of adopting LGD / OSM, duplicates of classes due to notation fluctuation of tags and the low coverage rate of actual facilities at the instance level could be a big problem. However, we think this approach is the best for our purpose in our best knowledge, because the LGD / OSM is the largest facility data that can be freely used at the present. Also note that actually it is impossible to measure how much the existing facilities are covered in reality. Regarding duplicates of classes in LGD, we select an entity as primary class that has both the most information-rich descriptions on the OSM and a large number of instances, then the rest are associated with owl:equivalentClass to the primary class.

The following shows an example of zoo class in the facility ontology. The meanings of Japanese words are added here in English as turtle comments for readers. Both a service of "see animal" and "pay admission fee for cultural facility" are actually described in the service ontology as subclasses of "see" service and "admission-viewing-gaming" service. Note that each service is described as a pair of an action and an action target, which users can perform. In this paper, we manually acquired and created service knowledge of facilities within the scenarios as necessary. See the statistic numbers in Table 1. As shown below, the lgdo:Zoo class is linked to the dbo:Zoo class in DBpedia Ontology to make possible to search related facility instances in DBpedia Japanese. The dbo:Zoo already has a link to Wikidata's wikidata:Q43501. Thus, it can be easily expanded when Wikidata is added.

```
lgdo:Zoo a owl:Class;
  servicevoc:dbpediaClass dbo:Zoo ;
  servicevoc:provideService [ servicevoc:hasService [
      servicevoc:action action:払う;                  # pay
      servicevoc:target target:文化施設入場料 ],  [  # admission fee
                                                 # for cultural facility
```

```
    servicevoc:action action:見る;                # see
    servicevoc:target target:動物 ]] ;            # animal
  rdfs:subClassOf servicevoc:Facility .
```

For the sake of systematical description of actions and action targets, we used the Household Income Balance Item Classification List (January, 2015) of the Statistics Bureau of the Ministry of Internal Affairs and Communications, of which items of statistics data are used to describe purchasing behavior at facilities. User's behavior at facilities can be divided into purchasing behavior (such as buying something or paying for some benefits as service) and the other actions (see, eat, drink, etc.). This classification is based on a hierarchical structure of action targets as users' behavior as consumer, so it is possible to consider cooperation with statistical data in future, starting with purchase actions. For actions and action targets other than purchasing behavior, we used Japanese WordNet, because we want to use WordNet's knowledge on the relationship between each verb as action and each noun as an action target. For instance, we made Action Target Ontology as follows.

```
target:動物 rdfs:label "動物";                  # animal
  servicevoc:wordnet wnja11instances:word-動物 .

target:食料 a owl:Class; rdfs:label "食料";     # food
  servicevoc:wordnet wnja11instances:word-食料 ;
  rdfs:subClassOf target:購買対象 .             # purchase object
```

The service ontology at the bottom of the table is the ontology we constructed this time, as explained in the above.

In the facility ontology, a number of services corresponding to distinct facilities come up with common abstract services. For example, both museums and art museums have the same service of "paying entrance fee for cultural facilities". In addition, there are hierarchical relationships among users' action targets, then we have a similar relationship between services. For example, "seeing animals" can be regarded as the top of "looking at a lion". We constructed an ontology of services apart from facility classes, so that services are independently recognizable, and it enabled us to expand the performance of inference by applying the hierarchy of services. In this paper, the part of service ontology is constructed by using the Classification in the Household Survey of the Ministry of Internal Affairs and Communications. The top of service ontology is the 'facility_service' and it is related to aspects of two types of behaviors, namely, 'purchase_service' focused on purchasing behavior, and an 'activity_service' focused on the other behaviors at facilities. The following shows an example of 'purchase_service' ontology entries.

```
  service:食料_サービス a owl:Class;        # food service
    rdfs:label "食料_サービス";
    servicevoc:action action:買う;         # buy
    servicevoc:target target:食料;         # food
```

Table 1. Outline of Prepared Datasets and Used Datasets

Dataset	Version	Num. triples	Num, classes	used
Fact Dataset				
DBpedia core+en	2016-04-01	1,131,657,931	-	△
DBpedia Japanese	2017-02-20	113,299,748	-	○
LinkedGeoData	2015-11-02	1,216,560,762	-	○
General Ontology				
DBpedia Ontology	2016-11-01	30,793	758	○
LGD Ontology	2014-09-09	24,530	1,200	○
Japanese WordNet	2013-06-26	4,003,288	57,238	○
Japanese Wikipedia Ontology	2013-11-07	21,863,327	166,397	×
YAGO	3.0.2	1,001,461,792	5,130,031	×
OpenCyc	2012-05-10	5,783,451	233,644	×
UMBEL	1.5	392,728	33,686	×
Service Ontology				
Facility Ontology	2017-02-20	3,257	418	○
Service Ontology	2017-02-20	3,933	750	○
Action Target Ontology	2017-02-20	2,030	622	○
Action Ontology	2017-02-20	153	55	○
subtotal of Service Ontologies		9,373	1,845	
Total		3,495,087,723	5,624,799	

```
  rdfs:subClassOf service:購買_サービス . # purchase service

service:肉類_サービス a owl:Class;        # meat service
  rdfs:label "肉類_サービス";
  servicevoc:action action:買う;          # buy
  servicevoc:target target:肉類;          # meat
  rdfs:subClassOf service:食料_サービス . # food service
```

4 Building Knowledge Graphs

We have collected a number of open knowledge resources as shown at the upper part of Table 1, and all of them are stored in one RDF store. However, at the time of this writing, we have actually used only DBpedia Japanese, LinkedGeoData, Japanese WordNet, and DBpedia Ontology as open datasets. Wikidata is not stored because of the capacity.

The system used one endpoint built with one dedicated RDF store.

5 Reasoning and Q&A Process

In this preliminary research, we process natural sentences only within the range expected at use-cases. Furthermore, in this paper it is assumed that the input is transcribed as text instead of speech.

5.1 Process Flow and Reasoning

Work flow of this system is as follows.

1. Input a text of user's requests.
2. Perform the morphological analysis for the input text.
3. Perform the case analysis starting with surface cases to deep cases.
4. Translate the requests into SPARQL queries.
5. Obtain the reply of SPARQL queries.
6. Generate the answering text from the obtained reply.

Japanese is a kind of agglutinative languages and a Japanese sentence is written without a space left among phrases and words. A noun phrase is composed of a noun and a particle, a verb phrase is composed of a stem of verb and a grammatical conjugation. So, morphological analysis is requisite in Japanese text processing in order to separate a sentence into phrases and words. Furthermore, particles attached to nouns decide the grammar case. For example, in response to an user's input "ライオンが見たいな (I want to see a lion)", the morphological analysis and *shift-reduce method* changes the Japanese sentence into the form of ((な (*pos info*) 8) ((たい (*pos info*) 6) (見 (*pos info*) 5)) ((が (*pos info*) 4) (ライオン (*pos info*) 0))), here (*pos info*) stands for a Part-of-Speech information of each, then case analysis produces the result such as Subject:NIL, Verb:(見る (*pos info*) 5), Object:(ライオン (*pos info*) 0), toPlace:NIL, fromPlace:NIL, Tool:NIL. Part-of-speech information obtained from morphological analysis is effectively used in various ways. For example, if there is an auxiliary verb 'たい (want)' next to a form of a behavioral verb such as '見る (see)' or '食べる (eat)', the whole sentence is interpreted as *request*. Thus, a request of seeing a lion is captured and transformed into a SPARQL query to the endpoints.

From the interpretation of request (see lion), the system searches facilities that can see a lion, using action target ontology and facility ontology. However, we have no common sense as LOD that a lion is in a zoo. When searching fails here, WordNet is used to generalize the target to more abstract ones by searching hypernym relations in WordNet until animal is found.

The SPARQL search picks up a number of facilities that are located near the current location, and the closest one to the current location is chosen outside of SPARQL search.

5.2 Inference with SPARQL

Initially, we attempted to make a plan by introducing IS-A logic function into planning based on classical state space reasoning and backward reasoning [1]. However, more than it, searching combined ontologies using one SPARQL query easily enabled us to retrieve acceptable instances of appropriate facility from the action target ontology and the facility ontology without any problems in execution speed. The LGD class according to the user's request from the facility ontology can be found, and once the LGD class is known, SPARQL allows direct retrieval of the facility instance within the LGD. If there is a DBpedia class linked from LGD, DBpedia Japanese is also automatically searched in SPARQL queries. The current system consists of RDF Store search and inference for interpretation of user's requests. This configuration is beneficial at usability and re-usability. Based on SPARQL search and open resources, it is possible to expand and refine ontology without touching the inference engine of the planning system in applications. It is meaningful for practical application of reasoning by large amount of data.

6 Example of Execution

The following shows an example of execution by this prototype system, see the added comments translated into English for readers.

```
SYSTEM(4): (eliza)
system> スポーツがしたいな。そのあと、温泉に行きたい。
;; I want to enjoy some sport, after that, I want to go to hot spring.
現在地はトヨタ東富士研究所です。
;; the current location is Toyota Higashfuji Institute.
スポーツをする場所を探します。; searching a location for sports
......
一番近くの場所を案内します。 ; guiding the nearest place
距離は 13.37621km です。     ; the distance is 13.37621km
場所：沼津市営球場           ; place: Numazu City Ball Park
緯度：35.1125                ; longitude
経度：138.863                ; latitude
URL："http://linkedgeodata.org/triplify/node2877270449"
現在地は(35.1125 . 138.863)です。; the current location is (35.1125 . 138.863)
温泉に入る場所を探します。   ; searching a location for hot spring
......
一番近くの場所を案内します。 ; guiding the nearest place
距離は 10.426165km です。    ; the distance is 10.426165km
場所：伊豆長岡温泉           ; place: Izu-Nagaoka Hot Spring
緯度：35.0353                ; longitude
経度：138.929                ; latitude
URL："http://ja.dbpedia.org/resource/伊豆長岡温泉"
```

Searching for a facility in the vicinity of the current location, the Toyota Higashifuji Institute, the system made a recommendation to go to Numazu City Ball Park, then go to Izu-Nagaoka Hot Spring, in response to a request to go to a hot spring after enjoying some sport.

While this prototype of action planning by using open knowledge sources and SPARQL queries is widely applicable to various kind of applications, yet there is not enough as intelligent agent. Making more intelligent agent remains in future work.

7 Discussion

In this preliminary research, the following issues are suggested.

1. It is necessary to understand data characteristics of coverage and granularity of each dataset, but it is generally hard for large datasets. At this time, we firstly made a utilization plan on the whole data set, after we examined the availability of actual data on the premise of these use-case scenarios.
2. Generally, it is tough work to find out correct relations between datasets. While simple string matching allows us an automatic matching process, the ontology mapping cannot be avoid human power at the present. While the accuracy of this mapping greatly affects the result, mechanical matching processing is difficult. In addition, we built intermediate ontologies and mapped them to LOD datasets, but building ontology is generally not easy for a novice.
3. Since DBpedia and LGD are datasets made by crowd sourcing, we cannot expect the completeness and validity of them. Missing or biased data is still problematic at reasoning. Actually, we found a closed food shop as results. At this time we attempted to eliminate errors as soon as it was found, but we need to think about some tools for (semi) automated error checking.
4. The inference procedure was designed according to these use-case scenarios. For other problems, different datasets and different work flows may be used. For example, it depends on features of a target problem about how the balance should be taken between general knowledge and fact data to solve the problem.

8 Conclusion

In this preliminary research, we made a prototype of action planning system for events of everyday life and world, based on open knowledge of LOD as fact data and taxonomy as common knowledge. We utilized a number of large-scale open databases and knowledge-bases. We found that we had already abundant knowledge about the everyday life and world as diverse open knowledge resources. This condition is very different at the era of Good-Old-Fashioned-AI (GOGAI) before the Web age and LOD. However, we also found that we needed the additional general and common knowledge that connects such different open resources in reasoning action plans with SPARQL endpoints. It is obvious that it will be necessary to make open knowledge more available not only in the verification and validation for each, but also in the combinations of them for applications.

References

1. Ghallab, M., Nau, D. and Traverso, P.: Automated Planning, theory and practice, Elsevier (2004) .