Multistep Expansion based Concept Search for Intelligent Exploration of Ontologies

Kouji KOZAKI
The Institute of Scientific and Industrial Research (ISIR), Osaka University
8-1 Mihogaoka, Ibaraki, Osaka
567-0047 Japan
+81-6-6879-8416
kozaki@ei.sanken.osaka-u.ac.jp

Yusaku KITAGAWA
The Institute of Scientific and Industrial Research (ISIR), Osaka University
8-1 Mihogaoka, Ibaraki, Osaka
567-0047 Japan
kitagawa@ei.sanken.osaka-u.ac.jp

ABSTRACT
Recently, there are various semantic data such as linked data, knowledge graph, ontologies and so on. How to explore them and get appropriate information are very important techniques for intelligent application systems based on them. In this article, we focus on intelligent exploration of ontologies since ontologies provide systematized knowledge to understand target domain and contribute to deep understanding of semantic data. Although concept search is very important and fundamental technique for ontology exploration, there are some rooms to improve it in order to deal with conceptual structure of ontologies more efficiently. The authors propose a novel conceptual search method called "Multistep Expansion based Concept Search" to get appropriate concepts from ontologies according to the user's intentions and purpose.

Categories and Subject Descriptors
I.2.4. [**Knowledge Representation Formalisms and Methods**]

General Terms
Management, Human Factors, Theory

Keywords
ontology, concept search, intelligent exploration

1. INTRODUCTION
Intelligent exploration of semantic data such as linked data, knowledge graph, ontologies and so on, in one of important techniques to make the most use of vast amounts of information being developed from day to day. In particular, we focus on intelligent exploration of ontologies since ontologies provide systematized knowledge to understands target world and contribute to deep understanding of semantic data.

In previous works, the authors proposed divergent exploration of ontology and showed it contribute to the user's understanding of ontology according to their intention and interests [1, 2]. Ontology explorations are done through finding concepts related with a focused concept and trace them repeatedly. When a ontology exploration system finds related concepts, it have to consider not only concepts which are directly connected with the focused concept through some relationships but also concepts which are indirectly related through conceptual structures in the target ontology. For example, when we explore a dish ontology in Fig.1, *meat* is directly connected with *meat dish* and indirectly related with *beef steak*. These related concepts are acquired by concept search techniques.

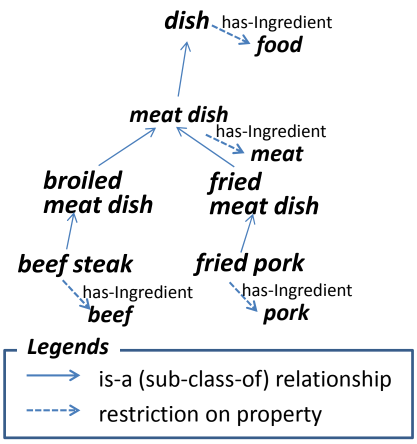

Fig.1 An example of dish ontology.

In order to get good results through ontology exploration, we need a technique of concept search to acquire appropriate concepts requested by the user. Though there are some approaches for concept search in ontologies, it seems that there are some rooms to improve it in order to deal with conceptual structure of ontologies more efficiently. It is partly because that most concept search techniques tend to get all concepts which satisfy search conditions while ontology explorations needs flexible classification of search result. To overcome these issues, the authors propose a novel conceptual search method called "Multistep Expansion based Concept Search" which the user can get appropriate concepts from ontologies according to the user's intentions and purpose.

This paper is organized as follows. The next section overviews related works about semantic search. Section 3 outlines basic characteristics of concept search in ontology. In Section 4, we propose Multistep Expansion based Concept Search and its implementation is discussed in Section 5. Finally, we present concluding remarks with a discussion of future work.

2. RELATED WORKS

There are many approaches for Semantic Search. They are classified into instance search and concept (class) search. Faceted Search is the most major method for instance searches. For example, Ferré proposes QFS (Query-based Faceted Search) which support to navigate faceted search by LISQL (Logical Information System Query Language) [3]. SPARQL[1] is the most major query language for Semantic Web. Because it is a query language for RDF, it also can be used for concept searches in ontologies which are written in RDF such as RDFS and OWL. However, SPARQL is not for concept search since it does not support semantics of ontology language. For example, when we want to get all sub classes of a class using SPARQL, we have to repeat queries to get classes which have *rdfs:subClassOf* properties.

On the other hands, some systems support concept search more efficiently. Protégé 4[2] provides DL Query tab[3] for concept search. It enables the user to search concepts whose definitions match a search condition described by the user in Manchester OWL Syntax[4]. This is a fundamental method for concept search.

Concept search are used for semantic search systems which use semantic information of ontologies. Dimitrios formalizes queries for concept search and develop a support system for DL query description using a graphical user interface [4]. This is an approach to help the user to describe queries. Popov proposes exploratory search called Multi-Pivot [5]. It extracts concepts and relationships from ontologies according to a user's interest. They are visualized and used for semantic searches. This is a good example, conceptual structures in ontology are used for semantic searches for instances. Ravish proposes a hybrid search which combines keyword search by natural language and semantic search [6]. PowerAqua[5] is a multi-ontology based question answering system [7]. They are also conceptual structure in ontologies, and concept search is one of key techniques to realize them.

Our approach shares a lot with concept search methods used in these existing semantic search system. However, our main concern is not how to query and get search results but how to classify search results. Many concept search systems show the search results just a list and some others show them with some categorizations in tag style or hierarchy structure. Although *is-a* hierarchies of concepts are used to show such categories, we want to introduce other classification of search results from ontological point of view.

3. CONCEPT SEARCH IN ONTOLOGY

In ontologies, concepts are defined by relationships with other concepts. Therefore, concept search in an ontology is not simple string matching but semantic search using relationships between them. The most essential relationships are *is-a* (sub class of) relationships between sub concepts (classes) and super concept (classes). Because *is-a* relationships have a semantics that sub concepts inherit all definition of their super concept, they are used to systematize concepts which have some similar definitions. Other relationships are used to represent definitions of concepts. They are used to describe search conditions for concept search.

For example, when a user searches "*dish* whose *ingredient* is *meat*" in a dish ontology shown in Fig.1, a search system find concepts which has *has-Ingredient* relationship (property) with *meat*. In the case of ontology in OWL, it means to find sub classes of *dish* which has a restriction on *has-ingredient* property as *meat*.

Here, according to semantics of *is-a* relationships discussed the above, all sub concepts (classes) of concepts which satisfy the search condition also satisfy it. That is, they also become the search result. However, it is depends on the user's intention and purpose which concepts are needed as the search result. In the case of the above example, we can consider some kinds of purposes as follows;

1) When the user wants to get the most general definition of dish whose ingredient is meat in order to know common characteristics of it, the search result should be only *meat dish*.
2) When the user wants to know dishes common to all kind of meat (not specific kinds of meat), the search result should be *meat dish*, *broiled meat dish* and *fried meat dish* whose ingredients are (not specific kinds of) *meat*.
3) When the user wants to know all dish whose ingredient is *meat*, the search result should be all *meat dishes* shown in Fig.1.

In order to deal with such intentions and purpose for concept search, we have to consider ontological meanings of concept search and classifications of search result. This topic is discussed in the following sections.

4. MULTISTEP EXPANSION BASED CONCEPT SEARCH

4.1 Basic Idea

In most semantic concept search methods, main search condition is definitions of concepts in which the user is interested. That is, its results are concepts whose definitions satisfy the search condition. Here, all sub concepts (sub classes) of the resulting concept also satisfy the search condition because all sub concepts inherit definition of their upper concepts (super classes). Therefore, it is a useful way to browse the search results one by one according to there *is-a* hierarchy.

In this method, though the search result is considered as a single set of concepts whose definitions satisfy the search condition, conceptual differences among these concepts are unclear.

For example, when a user searches "*dish* whose ingredient is *meat*" in an ontology shown in Fig.1, it is not clear whether its result should include "*dish* whose ingredient is *beef* or *pork*" or it should be only "*dish* whose ingredient is (not specific kinds of) *meat*". It depends on intention of the user. So, in order to represent the user's intention appropriately, the search result should be classified systemically.

It is important to systematize search results according to feature of concepts on which the user focuses so that the user can use conceptual definition in ontology efficiently. On the basis of this observation, the authors propose a novel search method, which is

[1] http://www.w3.org/TR/rdf-sparql-query/

[2] http://protegewiki.stanford.edu/

[3] http://protegewiki.stanford.edu/wiki/DLQueryTab

[4] http://www.co-ode.org/resources/reference/manchester_syntax/

[5] http://technologies.kmi.open.ac.uk/aqualog/okdeliverable/

named *Multistep Expansion based Concept Search*, which extracts concepts in ontology according to the user's interests represented as search condition.

The proposed method is consists of two expansion methods for concept hierarchies. One is *result expansion* which use *is-a* hierarchies of resulting concepts. The other is *condition expansion* which use *is-a* hierarchies of concepts which appear in search conditions. We can systematize search result in detail based on combinations of *result expansions* and *condition expansions*. In the following sections, we discuss classification of search result with features of concepts in each classification.

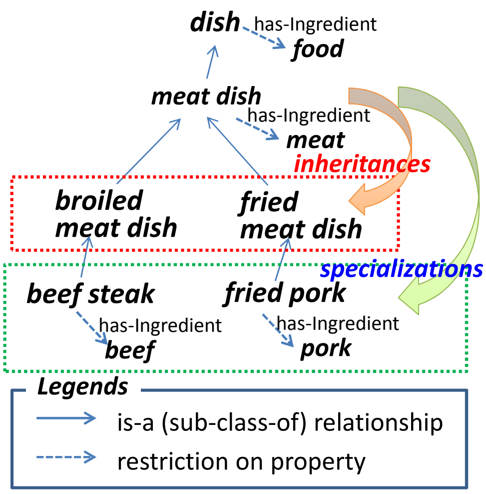

Fig.2 Examples of inheritances and specializations of restrictions

4.1.1 Result Expansion

In concept search in ontologies, its results are concepts whose definition satisfies its search condition. In the followings, we call the results *resulting concepts*. All sub concepts of a resulting concept also satisfy the search condition since all sub concepts inherit definitions from its super concept according to *is-a* hierarchy of them. Based on this feature of *is-a* hierarchy, we can expand a resulting concept according to its *is-a* hierarchy and get its sub concepts as another resulting concepts. We call such a method *result expansion*.

For example, when a user searches "*dish* whose ingredient is *meat*" and gets *meat dish* as a resulting concept, the user also gets its sub concepts such as *broiled meat dish* and *fried meat dish* as resulting concepts using *result expansions*.

Here, we can classify *result expansions* into three levels by considering inheritances and specializations of restrictions on properties. By specializations of restrictions we mean restrictions which specializes a restriction inherited from its super concept. In the case of an example in Fig. 2, *meat dish* has a restriction on *has-ingredient* property as *mea*. Its sub concepts such as *broiled meat dish* and *fried meat dish* inherit the restriction on *has-ingredient* as is. Furthermore, in their sub concepts such as *beef steak* and *fried pork*, restrictions on *has-ingredient* are specialized from *meat* to *beef* and *pork*. Note here that all of these sub concepts satisfy definition inherited form their super concept, in this case *meat dish*, whether some restrictions are specialized or not. Therefore, when a user searches "*dish* whose ingredient is *meat*", all of them satisfy the search condition.

On the basis of the above considerations, we introduce three levels of result expansions as follows;

<u>Result expansion: level 0</u>

- The user does not use result expansion and gets only top (the most upper) concept which satisfy a search condition as resulting concepts.

<u>Result expansion: level 1</u>

- The user gets resulting concepts which inherit the restriction on property specified in a search condition as is without specializations. That is, when the restriction is specialized in sub concepts, they are not expanded.

<u>Result expansion: level 2</u>

- The user gets all resulting concepts inherit the restriction on property specified in a search condition whether it is specialized or not. That is, all sub concepts of the resulting concepts gotten in level 0 are expanded.

In the case when a user searches "*dish* whose ingredient is *meat*" in an ontology shown in Fig.2, resulting concepts according to result expansions are as follows;

<u>Result expansion: level 0</u>

- *meat dish*

<u>Result expansion: level 1</u>

- *meat dish*, *broiled meat dish* and *fried meat dish*

<u>Result expansion: level 2</u>

- *meat dish*, *broiled meat dish* and *fried meat dish, beef steak and fried pork*

Here, we call concepts whose definition have a restriction specified as the search condition *search condition directly match concepts*. And we call concepts whose definition have a specialization of the restriction specified as the search condition is defined *specialized search condition match concepts*.

4.1.2 Condition Expansion

When a user search concepts whose definition refers a concept (denoted by *C*) in a property restriction, concepts whose

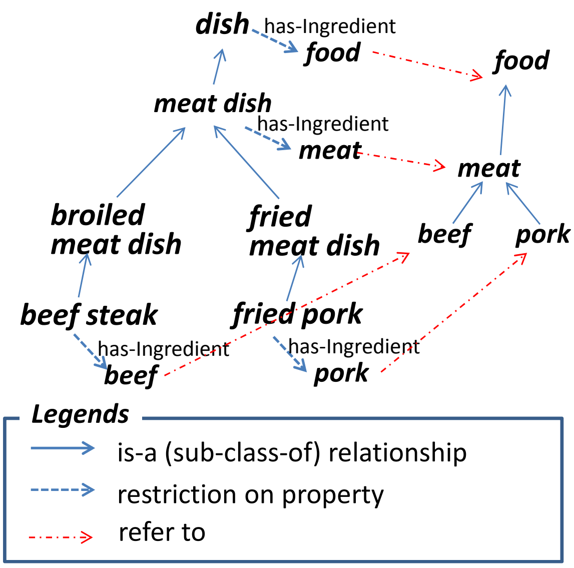

Fig.3 A dish ontology.

definition refers a sub concepts of C also satisfy the search condition. That is, when a search condition is "concepts whose definitions include a restriction on property p as concept C", "concepts whose definitions include a restriction on property p as sub concepts of C" are also its resulting concepts. In this way, we can expand a concept which is referred to in search condition according to its *is-a* hierarchy and get its concepts whose definitions refer to its sub concepts as another resulting concepts. We call such a method *condition expansion*.

For example, when a user searches "*dish* whose ingredient is *meat*" in a dish ontology shown in Fig.3, its search condition is "*dish* whose definition includes a restriction on *has-ingredient* property as *meat*". In a case that the user does not use condition expansion, he/she gets only *meat dish,* which has a restriction on *has-ingredient* property as *meat*, as the search result. On the other hand, in a case that the user use condition expansions, the search condition is expanded to "*dish* whose definition includes a restriction on *has-ingredient* property as *beef* or *pork*" according to is-a hierarchy of *meat*, and the user gets *beef steak* and *fried pork* as the search result.

As discussed in 3.1.1, we call concepts in which a restriction specified as the search condition is defined (e.g. *meat dish*) *search condition directly match concepts*, and we call concepts in which a specialization of the restriction specified as the search condition is defined (e.g. *beef steak* and *fried pork*) *specialized search condition match concepts*.

When we use condition expansions, we do not consider *is-a* hierarchies of resulting concepts. That is, resulting concepts by condition expansions do not include concepts which inherit the restriction on property specified in a search condition as is. It means that resulting concepts by condition expansions represent boundary where definition specified in a search condition is specialized.

4.2 Combination of Result Expansion and Condition Expansion

4.2.1 Differences of Resulting Concepts

Before we discuss combination of result expansion and condition expansion shown in Fig.4, we consider differences of resulting concepts by two expansions. When we use result expansion, all resulting concepts are sub concepts of concepts which directly satisfy an original search condition. On the other hand, when we use condition expansion, search condition is expanded according to *is-a* hierarchy of a concept which is referred to in an original search condition. Therefore, resulting concepts by condition expansion could include concepts other than resulting concepts by result expansion.

For example, we suppose that a user search "*dish* whose *ingredient* is *meat*" in a dish ontology shown in Fig.5 (its search condition is "*dish* whose definition includes a restriction on *has-Ingredient* property as *meat*"). In this case, the user gets *meat dish* as a *search condition directly match concepts*. Then, the user gets sub concepts of *meat dish* (e.g. *beef steak*, *fried pork*) as resulting concepts by result expansions. On the other hand, the user can also get *beef salad* and *pork salad* as resulting concept by condition expansions while they are not included in resulting concepts by result expansions.

In this way, resulting concepts by result expansions and condition expansions are different It suggests us that combinations of two expansions more detailed classification of search result.

4.2.2 Patterns of Combinations

In this section, we discuss combinations of result expansions and condition expansions. Because result expansion has three levels and condition expansion has two levels (apply or not), we can consider six patterns of combinations as follows;

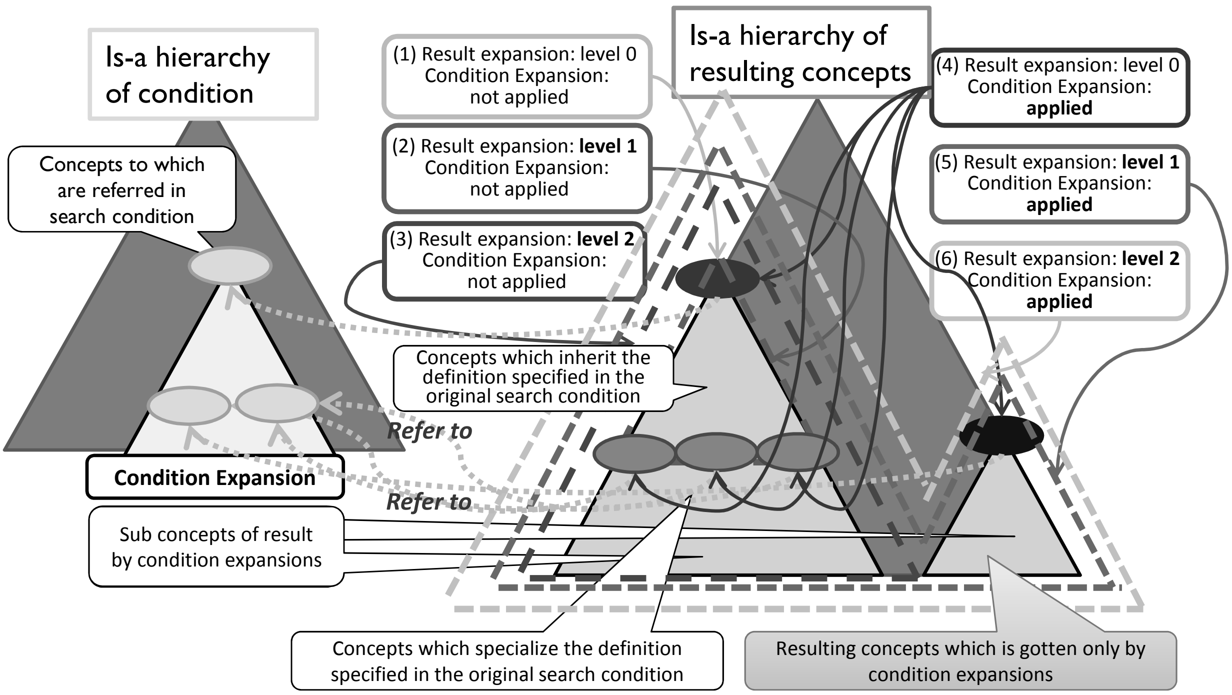

Fig.4 A classification of ranges of resulting concepts by combination of result expansion and condition expansion.

(1) Result expansion: level 0 + Condition Expansion: not applied

(2) Result expansion: **level 1** + Condition Expansion: not applied

(3) Result expansion: **level 2** + Condition Expansion: not applied

(4) Result expansion: level 0 + Condition Expansion: **applied**

(5) Result expansion: **level 1** + Condition Expansion: **applied**

(6) Result expansion: **level 2** + Condition Expansion: **applied**

Fig. 4 shows the classification of ranges of resulting concepts by each pattern. In the followings, we discuss each pattern using an example search of "*dish* whose definition includes a restriction on *has-Ingredient* property as *meat*" in the dish ontology shown in Fig.5.

Pattern (1) is concept search without result expansions nor condition expansion. Its resulting concept are only *search condition directly match concepts* (e.g. *meat dish*).

Pattern (2) and (3) get sub concepts of *search condition directly match concepts*, which are result by (1), as resulting concepts. In the case of (2), resulting concepts are concepts which inherit the definition specified in the original search condition as is (e.g. *broiled meat dish*, *fried meat dish*). In the case of (3), resulting concepts include concepts whose definitions are specialized from the definition specified in the original search condition (e.g. *beef steak*, *fried pork*). That is, resulting concepts by pattern (3) are all sub concepts of resulting concept by (1).

Pattern (4) gets not only *search condition directly match concepts* (e.g. *meat dish*) but also *specialized search condition match concepts*. Some of them (e.g. *beef salad* and *pork salad*) are not sub concepts of the *search condition directly match concepts* as discussed in 3.2.1.

Pattern (5) and (6) also gets sub concepts of result by (4) as resulting concepts. Although pattern (5) expands only sub concepts which inherit the definition specified in expanded search conditions by (5) as is, its resulting concepts are the same with result by (6) because *specialized search condition match concepts* are already expanded by (4). That is, (5) and (6) can be integrated to one pattern and its resulting concepts include all concepts which satisfy the original search condition. Furthermore, when all *specialized search condition match concepts* are sub concepts of *search condition directly match concepts*, resulting concepts by (3), (5) and (6) are the same.

4.2.3 Classification of Resulting Concepts

In order to systematize result of concept search, we consider classification of resulting concepts by combinations of result expansions and condition expansions. All resulting concepts are considered as subset of *is-a* trees because of characteristics of *is-a* relationships which all sub concepts inherits definition of their super concepts. So, we discuss how the sub trees of resulting concepts are classified according to ontological characteristics of them.

When we consider patterns of combinations of result expansions and condition expansions, resulting concepts are classified into sub concepts of *search condition directly match concepts* by pattern (1) or (2) and sub concepts of *specialized search condition match concepts* by others. S*pecialized search condition match concepts* and sub concepts of them are classified according to whether they are sub concepts of *search condition directly match concepts* or not. We call the former *directly type* and the latter *distributed type*. As a result, we can classify resulting concept by multistep expansion based concept search as follows;

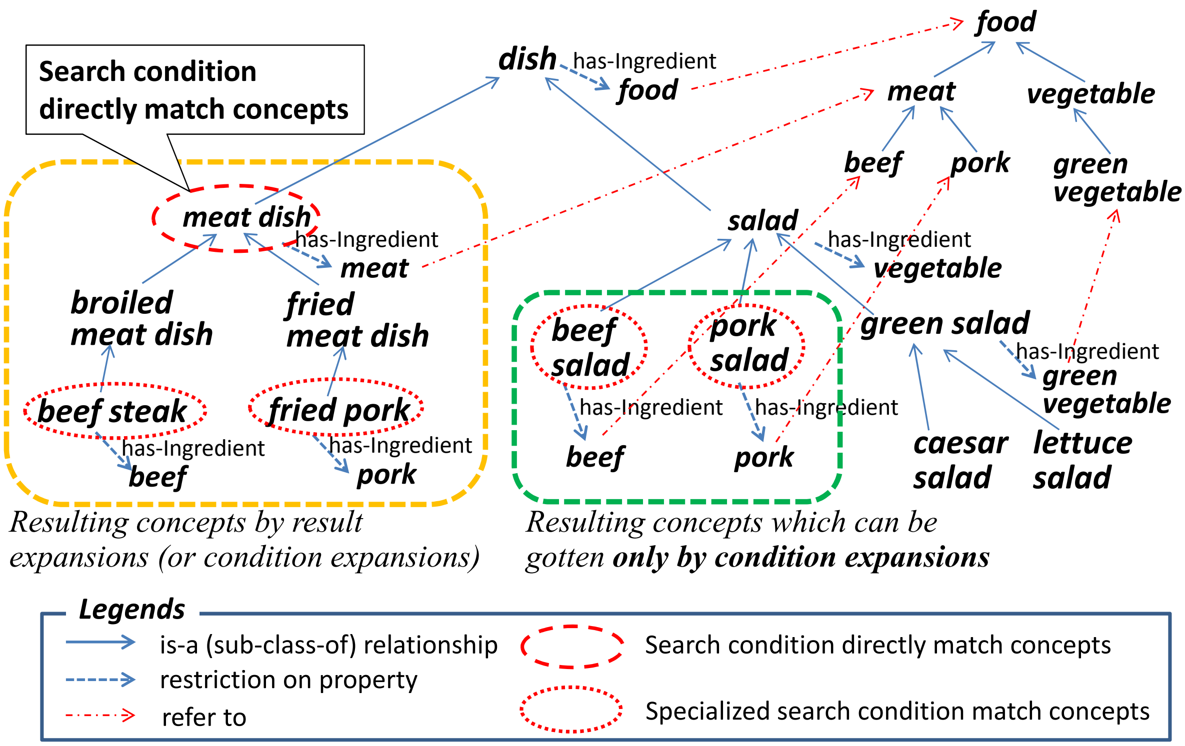

Fig.5 Differences of resulting concepts by result expansions and condition expansions.

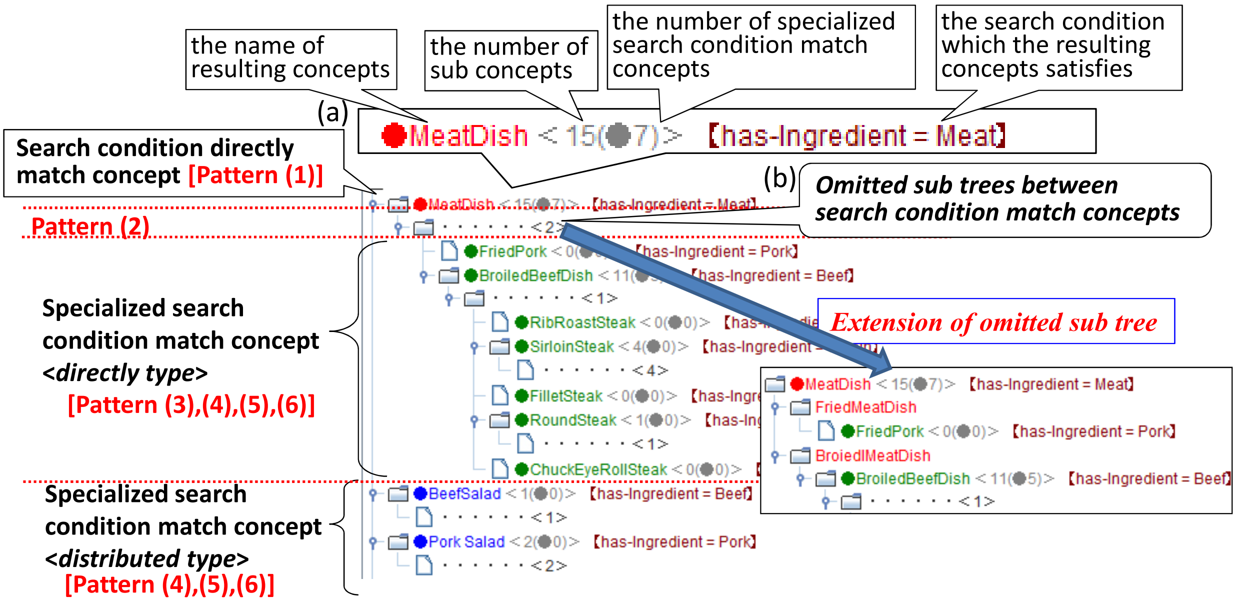

Fig.6 An example of search result when a user searched "*dish* whose definition includes a restriction on *has-Ingredient* property as *meat*" in Multistep Expansion based Concept Search System.

(a) ***Search condition directly match sub tree***

Is-a trees of resulting concepts whose definitions satisfy an original search condition without specializations. They represent the most general concepts which satisfy the search condition (it corresponds to definitions focused by a user). *Search condition directly match concepts* are representing concepts of them and become the root concepts of this sub trees.

(b) ***Specialized search condition match sub tree***

Is-a trees of resulting concepts whose definitions satisfy specialized search condition by result expansions. They represent how the search condition, which corresponds to definitions focused by a user, is specialized according to *is-a* hierarchies. *Specialized search condition match concepts* are representing concepts of them and become the root concepts of this sub trees.

Specialized search condition match sub trees are classified into following two types;

(b1) ***Specialized search condition match sub tree <directly type*** **>** is a sub tree of type (a).

(b2) ***Specialized search condition match sub tree <distributed type*** **>** is not a sub tree of type (a).

5. IMPLEMENTATION

5.1 Multistep Expansion based Concept Search System

The authors developed a Multistep Expansion based Concept Search System based on the considerations in section 3. Fig.6 shows an example of search result when a user searched "*dish* whose definition includes a restriction on *has-Ingredient* property as *meat*" in a dish ontology (It is another ontology than the dish ontology in Fig.5).

The search result is visualized using graphical user interface in a tree. Resulting concepts are represented by tree nodes[6] according to classifications of resulting concepts discussed in 3.2.3 as follows;

(a) Red label nodes: Search condition directly match sub tree

- **Red icon:** Search condition directly match concepts

(b1) Green label nodes: Specialized search condition match sub tree <directly type >

- **Green icon:** Specialized search condition match concepts <directly type >

(b2) Blue label nodes: Specialized search condition match sub tree <distributed type >

- **Blue icon:** Specialized search condition match concepts <distributed type >

A Node with icon shows the name of resulting concepts, the number of its sub concepts, the number of specialized search condition match concepts in its sub concepts, the search condition which the resulting concepts satisfies (see Fig.6(a)). Nodes whose label is "・・・・・" represents sub trees between search condition match concepts is omitted to show. The number with the node represents the number of omitted concepts. When the user clicks the node, the omitted concepts are shown like Fig.6(b).

Through this tree interface, the user can capture and understand results of concept search systematically according to definition he/she has interested in while it is difficult in existing one by one expansions of *is-a* tree.

---

[6] We do not consider the color scheme used for the implemented system yet. We will improve the design of icons and their color in the future since this is important from an HCI point of view.

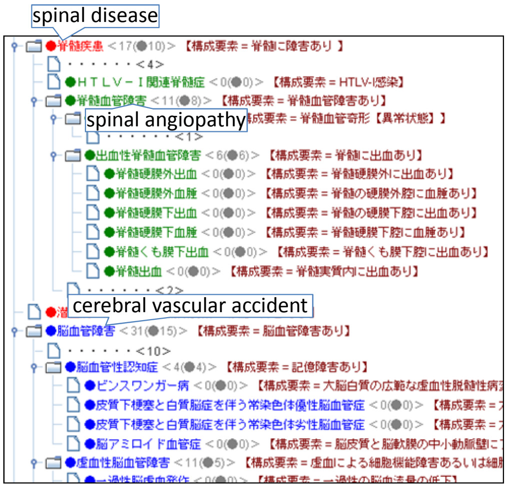

Fig.7 An example of search result in a disease ontology.

5.2 Application to Disease Ontology

We applied the proposed system to search disease concepts in a disease ontology discussed in [8]. In the disease ontology, about 6,000 diseases are defined as causal chains of abnormal states. That is, each disease is constituted of abnormal state. Thanks to such definition, we can search diseases according to which abnormal state constitutes them. Fig.7 shows an example of search result in the disease ontology. Its search condition is "diseases whose constituents include *myelopathy*". The search result shows us that its resulting concepts are classified according to the classifications discussed in 3.2.3. It can help us to understand the search result systematically particularly when we search concepts in a large scale ontology such like the disease ontology.

6. CONCLUSION

In this paper, we proposed Multistep Expansion based Concept Search and a search system based on the method. Its feature is classifications of search results according to ontological differences of resulting concepts. It could help the users to capture and understand the search result according to their interests. As the result, it could be an essential technique for semantic exploration of ontologies and other semantic data based on ontologies.

Future works include an integration of the proposed method and other ontology exploration methods such like our previous works [1, 2], application of the system for other ontologies, and improvements of the system through feedbacks from them.

7. ACKNOWLEDGMENTS

This work was supported by JSPS KAKENHI Grant Numbers 24120002 and 22240011.

8. REFERENCES

[1] Kozaki K, Hirota T and Mizoguchi R, Understanding an Ontology through Divergent Exploration, Proc. of ESWC 2011, Part I, LNCS 6643, pp.305-320 (2011).

[2] Kozaki K, Saito O and Mizoguchi R, A Consensus-Building Support System based on Ontology Exploration, Proc. of IESD, (2012)

[3] Sébastien Ferré, Alice Hermann: Semantic Search: Reconciling Expressive Querying and Exploratory Search, In Proc. of ISWC, LNCS7031, (2011), 177-192.

[4] Dimitrios A. Koutsomitropoulos, Ricardo Borillo Domenech, Georgia D. Solomou: A Structured Semantic Query Interface for Reasoning-Based Search and Retrieval, In Proc. of ESWC, LNCS6643, (2011), 17-31.

[5] Igor O. Popov, m.c. schraefel, Wendy Hall, Nigel Shadbolt: Connecting the Dots: A Multi-pivot Approach to Data Exploration, In Proc. of ISWC, LNCS7031, (2011), 553-568.

[6] Ravish Bhagdev, Sam Chapman, Fabio Ciravegna, Vitaveska Lanfranchi, Daniela Petrelli: Hybrid Search: Effectively Combining Keywords and Semantic Searches. ESWC 2008 (2008), 554-568.

[7] PowerAqua: Fishing the Semantic Web. Vanessa Lopez, Enrico Motta, Victoria Uren. Proceedings of the European Semantic Web Conference, ESWC 2006, Montenegro.

Mizoguchi R, Kozaki K, Kou H, Yamagata Y, Imai T and Ohe K, River Flow Model of Diseases, In Proc. of ICBO 2011, Buffalo, NY, July 28 – 30 (2011), 63-70.