Learning a Large Scale of Ontology from Japanese Wikipedia

Susumu Tamagawa*, Shinya Sakurai*, Takuya Tejima*, Takeshi Morita*, Noriaki Izumi† and Takahira Yamaguchi*

*Keio University
3-14-1 Hiyoshi, Kohoku-ku, Yokohama-shi, Kanagawa 223-8522, Japan
Email: {s_tamagawa, yamaguti}@ae.keio.ac.jp
†National Institute of Advanced Industrial Science and Technology
2-3-26 Aomi, Koto-ku, Tokyo 135-0064, Japan

Abstract—Here is discussed how to learn a large scale of ontology from Japanese Wikipedia. The learned ontology includes the following properties: rdfs:subClassOf (IS-A relationships), rdfs:property (class-instance relationships), owl:Object/DatatypeProperty (Infobox triples), rdfs:domain (property domains), and skos:altLabel (synonyms). Experimental case studies show us that the learned Japanese Wikipedia Ontology goes better than already existing general linguistic ontologies, such as EDR and Japanese WordNet, from the points of building costs and structure information richness.

I. INTRODUCTION

It is useful to build up large-scale ontologies for information searching and data integration. Among the large-scale ontologies are WordNet [1] and Cyc [2]. However, it takes many costs to build these ontologies by hands. Moreover, the manual ontology engineering process makes many bugs come up and maintenance and update difficult. For these reasons, more attention comes to build (semi) automatic ontologies on research, ontology learning.

Wikipedia, the Web-based open encyclopedia, becomes popular as new information resources [3]. Since Wikipedia has rich vocabulary, good updatability, and semi-structuredness, there is less differences between Wikipedia and ontologies when compared with free text. Thus, ontology learning from Wikipedia becomes popular. This paper proposes and evaluates a large-scale, general-purpose ontology learning method using the Japanese Wikipedia as resources.

This paper is structured as follows: We introduced related work about deriving ontology from Wikipedia in Section II. In Section III we explain what Wikipedia Ontology we developed is and the detailed extraction techniques to Wikipedia. In Section IV we showed the result of experiment that we actually applied the extraction techniques to Wikipedia. In Section V we showed overall view and characteristics of the Wikipedia Ontology. In Section VI we introduced search tool named “WiLD” using Linked Data as ways of utilizing the Wikipedia Ontology. Finally we present conclusion of this paper and our future work.

II. RELATED WORK

Auer et al.’s DBpedia [4] constructed a large information database to extract RDF from semi-structured information resources of Wikipedia. They used information resource such as infobox, external link, categories the article belongs to. However, properties and classes are built manually and there are only 170 classes in the DBpedia.

Fabian et al. [5] proposed YAGO which enhanced WordNet by using the Conceptual Category. Conceptual Category is a category of Wikipedia English whose head part has the form of plural noun like American singers of German origin. They defined a Conceptual Category as a class, and defined the articles which belong to the Conceptual Category as instances of the class. As far as instances concerned, they defined metadata using their own property such as bornInYear and locatedIn. However, this method cannot be applied to the Japanese language due to the fact that the techniques depend on English grammar. In this paper we propose the method which can define much instance information and are available to Wikipedia Japan. As for property, they identified frequent 170 attribute in infoboxes. Moreover they matched identified attribute and their own properties, and defined domain and range information. For example, the attribute of born and birth are matched to birthDate property whose domain is Person and range is TimeInterval.

Link mining is a newly emerging research area for the big network of the current WWW. This is at the intersection of the work in link analysis, hypertext and web mining, relational learning and inductive logic programming, and graph mining. In link mining, there are Object-Related tasks, Link-Related Tasks, Graph-Related Tasks according to the survey by Getoor et al. [6]. As for Wikipedia, Nakayama et al. [3] has proposed Wikipedia Mining which is a mining for Wikipedia. They have mentioned the following characteristics of Wikipedia: Dense link structure, URL as an identifier, Brief link texts, and Live update.

Ponzetto et al. [7] tried to extract is-a relationship from category tree of Wikipedia. They proposed six methods to extract various relationships. The main method of these called Syntax-based methods is the simple matching method.

978-0-7695-4191-4/10 © 2010 IEEE
DOI 10.1109/WI-IAT.2010.177
III. BUILDING THE WIKIPEDIA ONTOLOGY

The Wikipedia Ontology is built from the following five types of relationships. Note, however, that the relationships enclosed in parentheses are vocabularies (classes and properties) defined by OWL\(^1\), RDFS\(^2\), RDF\(^3\), SKOS\(^4\), each of which corresponds to extracted relationships. The relationship building methods are introduced briefly in this section.

A. Is-a relationship (rdfs:subClassOf)
B. Class instance relationship (rdf:type)
C. Infobox triple (owl:Object/DatatypeProperty)
D. Property definition domain (rdfs:domain)
E. Synonyms (skos:altLabel)

A. Building Is-a Relationships

Although Wikipedia has hierarchical categories, their objective is to categorize articles. Since the relationship of the lower level categories and higher level categories is often not seen in terms of is-a relationships viewed from the perspective of inheriting a quality, it is difficult to utilize the Wikipedia category hierarchy with the is-a relationships without refinement. As a result, the is-a relationship is built by the following three methods:

1. Matching of the character string related to the category hierarchy
2. Matching of the category name and infobox template
3. Scraping of the table of contents headings

1) Matching of the character string related to the category hierarchy: The character string matching for the category hierarchy utilizes two methods, one called Backward String Matching, the other called Forward Matched String Eliminating. In English, we can easily split meaningful terms from continuous array of characters with spaces. However, in Japanese, it is difficult to split meaningful terms from characters because of usually no space in Japanese characters. Therefore, we utilize character string matching methods to split meaningful terms from characters in the category tree. Although these are language-specific techniques, they are effective for extracting is-a relationships from Japanese Wikipedia category tree.

The method of Backward String Matching extracts category links whose sub category’s string is the form of “arbitrary string + superior category name”. We regard such links as is-a relationship, and extract them. Figure 1 shows an example of this method.

We’ll show the example of extracting is-a relationship by using Forward Matched String Eliminating. This is the case when there are one arbitrary category (Category 1) and its sub-category (Category 2). When Category 1 is the form of “A + arbitrary string” and Category 2 is the form of “A + other arbitrary string” (A represents the same word.), the method extract is-a relationship by eliminating both of A. Figure 2 shows an example of this method.

2) Matching of the category name and infobox template: The table called infobox often exists in the article of Wikipedia, and it shows article’s information in the form of a set of attribute-value rows. As far as infobox is concerned, there is a feature that sets of attributes are prepared according to domain as templates in advance. For example, attributes of “developer”, “platform” etc. defined in the “programming language” template.

We focus on the relation among the infobox templates, categories to which the articles which have a infobox belong (infobox’s categories), and category tree. Then we propose how to extract is-a relationship by matching infobox templates name to category names. Figure 3 shows an example, and the procedure of this method is shown following a - d. Using this method, we could extract is-a relationships which we could not extract by the method we explained at Matching of the character string related to the category tree.
hierarchy.

a. Extract the infobox templates, infobox’s categories, and category tree from Wikipedia dump data.
b. Match infobox templates name to category names in the category tree.
c. Extract sub-categories of the matched categories at b. and trim the categories other than infobox’s categories from the sub-categories.
d. Define is-a relationships between the extracted categories at b. and c.. (The extracted categories at c. as sub classes and the extracted categories at b. as super classes.)

3) Scraping of the table of contents headings: The Wikipedia article has a table of contents. This method focuses on the fact that it is easy to extract is-a relationships when the terms “class” and “type” are included in the table of contents of Japanese Wikipedia articles. The is-a relationships are extracted by scraping articles that have tables of content that include the terms “class” and “type”. Figure 4 shows an example.

In the article on Asteroid spectral types is the term “Present-day classifications” in the table of content. After that is a hierarchical description, and with “Tholen classification” and “SMASS classification” as discriminatory attributes, terms such as “C-group” are listed. These build is-a relationships with Asteroid spectral types as the highest level concept.

B. Building Class-Instance Relationships

In listing page of Wikipedia various things names are enumerated. For example, many world’s language names are listed in “Language listing page”. Since it isn’t necessarily for editor of Wikipedia to make an arrangement of articles and confirm the detail of facts, it is said that there are a lot of people who participate editing of listing pages. Therefore, there are many listing pages, and their reliability of accuracy is very high. So we considered that scraping the listing pages enables us to extract many class-instance relationships. We propose the scraping method to listing pages shown in following a - g.

All the article text is available from dump data page for free in the form of XML. First, eliminate text other than listing pages by using “page” tag, “title” tag and other tags like that. The string of instances is described in the lines beginning with “*” or “#” (we call them “*” lines from now.), and the string to be taxonomical attribute is described in the lines beginning with “=” (we call them content index.). So, scrape the lines beginning with other than “*”, “#” or “=”. At the followings b - f, eliminate “*” lines which don’t include correct instance’s strings with a certain pattern we developed.

a. Scrape lines including instances’ string
b. Eliminate * lines which are used in the explanatory text of the list.
c. Eliminate * lines which are linked to other listing pages.
d. Eliminate * lines which belong to unconcerned content index like “recital”.
e. Eliminate * lines which aren’t correct as instances like “* REDIRECT”.
f. Eliminate * lines which are used to describe year like “* 19th”.
g. Scrape the string of instance out of each “*” lines by using symbol of link “[[ ]]” to identify it.

C. Infobox Triple

The three sets that have an infobox, “articles, subjects and values,” can also be seen as three other sets, “instance, property, and property values.” Thus, the triple can be extracted from the infobox by scraping and using this structure. Further applying of the modeling rules to the 40 infobox templates makes the category owl:ObjectProperty/owl:DatatypeProperty possible as a property type.

Actually, the type of properties from infobox templates are defined as owl:ObjectProperty, when values of the properties are resources. On the other hand, the type of the properties are defined as owl:DatatypeProperty, when values of the properties are literals. The number of Infobox templates used was specified 40 in this case because the modeling of about 146,000 Infobox (about 72%) was able to be done with the occurrence rate high while the total of Infobox was about 202,000 when extracting it beforehand by the use of the Wikipedia dump data in October, 2009 and template of 40 kinds of the title.

D. Extracting the Property Domain

The subject of the infobox triple discussed in III-C is the article name as an instance. For that reason, seeking the category to which the article that is the subject belongs makes it possible to define the domain of the property. Thus, the infobox template name is extracted as the domain of each property that the infobox possesses. Figure 5 shows the example.
Moreover, the correct sub-category is correlated to each property domain held by the template as an is-a relationship that was extracted using the method in number (2) of III-A, called matching the category name and infobox template, which is done as the extraction of a property domain that has not been defined by a template.

E. Extracting Synonyms

Synonyms are extracted using Wikipedia’s redirect link function. The synonyms extracted in this case are correlated to each term of the corresponding class or instance extracted in III-A to III-B. We use SKOS property to describe synonymous relationships such as “skos:prefLabel” and “skos:altLabel”. SKOS provides a model for expressing the basic structure and the content of concept schemes. The “skos:prefLabel” property allows you to assign a preferred lexical label to resources. The “skos:altLabel” property allows you to assign an alternative lexical label to a concept.

IV. EXPERIMENTAL RESULTS AND OBSERVATIONS

We now discuss the results and observations from extracting Wikipedia dump data as a source as of December 2009 using the method in III

A. Results and Observations of the Is-a Relationship

1) Results and Observations of Matching of the character string related to the category hierarchy: We extracted a total of 12,558 is-a relationships, 7,971 through rear character strings and 4,587 through front character strings, relative to a total of 91,316 categories. We extracted 1,000 samples from the parent population of 12,558 and determined their truthfulness and falseness. We used following expression (1) as a formula for 95% confidence interval. In the formula (1), n represents the number of samples, N represents population, and \( \hat{p} \) represents the estimated amount which is the number of accuracy samples divided by total number of samples.

\[
\hat{p} - 1.96 \sqrt{\left(1 - \frac{\hat{p}}{n}\right) \frac{n \cdot \hat{p}}{n - 1}} \leq p \leq \hat{p} + 1.96 \sqrt{\left(1 - \frac{\hat{p}}{n}\right) \frac{n \cdot \hat{p}}{n - 1}}
\]  

(1)

The results had an accuracy rate of 93.1 ± 1.51%. The errors were due to things such as extracting class-instance relationships and has-a relationships as well as having an abstract parent class. An abstract class is a category that resides at the top of Wikipedia’s category hierarchy. As with the upper level ontologies, the category hierarchy of the Japanese Wikipedia does not have strict categories, but nine “main categories” that are the root categories. Therefore, many inappropriate relationships are deemed is-a relationships between these root categories and the categories immediately below them. The impact is that they are also extracted. Table I shows examples of extracted is-a relationships by the method of Backward String Matching and Forward Matched String Eliminating and Table II shows incorrect part of is-a relationship we extracted.

2) Results and Observations of Matching of the category name and infobox template: Of this method, We extracted 3,782 is-a relationships with an accuracy rate of 93.2 ± 1.34%. As for the recall rate, the result was 68.7%. Table III shows evaluation of the matches between the category names and infobox templates.

When we calculated recall of is-a relationships, we used as answer set sub-categories which has been judged to have correct is-a relationships to matched category by hand. There are a number of reasons for recall being low, but we believe the main reason is that the number of articles that have infoboxes (approximately 210,000) is low (approximately 30%) relative to the total number of Wikipedia articles (approximately 640,000). Thus, it is not possible to obtain a comprehensive grasp on the categories to which articles possessing an infobox belong.
3) Results and Observations of Scraping of the table of contents headings: We were able to extract as many as 83,288 is-a relationships by scraping the headings of the table of contents. Table IV shows examples of extracted is-a relationships by scraping the headings of the table of contents and we showed incorrect part of is-a relationship we extracted in Table V.

Here, the is-a relationships extracted were very large in terms of scale. One characteristic is that hierarchical relationships such as “sushi, rolled sushi, futomaki sushi” were also extracted. However, within the is-a relationships extracted by this method are things such as “piano, Other types”. It includes a class such as “other” which is dependent upon context. Taking context into account, as with “Other types of piano” some compensation needs to take place. The accuracy rate used to carry out this compensation is 86.1 ± 2.13%. It appears that most of the mistakes are due to the scraping rule being inadequate; so, a tighter rule needs to be added.

B. Results and Observations from Extracting Class-Instance Relationships

We obtained 323,024 instances; the number of classes generated from article names in the article list was 2,902; class-instance relationships amounted to 421,989. The accuracy of the class-instance relationship is high, 97.2 ± 1.02%. Table VI shows examples of extracted instances and Table VII shows incorrect part of class-instance relationships we extracted.

As an example of a class that has many instances, the instances of Japanese celebrities is overwhelmingly large, with 3,685 instances of Japanese voice actors/actresses, 3,180 instances of Japanese mountain path, 2,854 instances of Japanese cartoonists, 2,321 instances of Japanese male actors, and 2,144 instances of US Ship. The result means that Wikipedia Japan has more contents about human. At the same time our method could extract instances from broad domain like “Japanese mountain path” or “US Ship”.

We see from Table VII that most incorrect relationships are due to lack of rules of scraping. First fault and second row in Table VII represents lack of rule which eliminates language code like “:en:”, and parenthetic reference. Third and fourth row in Table VII represents lack of pattern which specifies string of instance in the “*” row. Other fault of relationships was having extracting Class-subclass relationships shown in fifth row in Table VII.

C. Results and Observations of Properties from Infobox Triple

We extracted 1,203,404 infobox triples. The types of properties in infobox triples amounted to 6,995. We were able to classify 69.3% of infobox triples through the modeling of 40 infobox templates. Table VIII shows properties with high utilization rates.

Under the infobox triple, many of the properties with high utilization rates were the owl:ObjectProperty. The owl:DatatypeProperty, such as the “year, month, birthDate” property and the “amount of capital” property was also extracted. The accuracy rate was 95.8 ± 1.79%. Within this, the infobox triple accuracy rate due to the ability to classify using modeling was high, 98.6%. Many of the errors were scraping errors. In particular, many scraping errors came about when the URL was described for the property value. Additionally, the property by fiscal year was often extracted. A lot of this cases are seen in the player category and the infobox has the result every fiscal year in the player article.

D. Extraction Results and Observations of the Property Domain

We were able to extract 9,367 property domains for the 6,995 properties extracted in IV-C. The accuracy rate was 95.4 ± 1.22%. We extracted property domains not defined

<table>
<thead>
<tr>
<th>Table III</th>
<th>EVALUATION OF EXTRACTED IS-A RELATIONSHIPS BY MATCHING OF THE CATEGORY NAME AND INFOBOX TEMPLATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Matched category</td>
<td>Sub category</td>
</tr>
<tr>
<td>soccer club</td>
<td>99</td>
</tr>
<tr>
<td>organic compound</td>
<td>76</td>
</tr>
<tr>
<td>tennis player</td>
<td>73</td>
</tr>
<tr>
<td>spa town in Japan</td>
<td>48</td>
</tr>
<tr>
<td>soccer player</td>
<td>144</td>
</tr>
<tr>
<td>inorganic compound</td>
<td>15</td>
</tr>
<tr>
<td>dam</td>
<td>14</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>average</td>
<td>17.82</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table IV</th>
<th>EXAMPLE OF IS-A RELATIONSHIPS BY SCRAPING OF THE TABLE OF CONTENTS HEADINGS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Super Class</td>
<td>Sub Class</td>
</tr>
<tr>
<td>Bryophyte</td>
<td>Sphagnidae</td>
</tr>
<tr>
<td>Chocolate</td>
<td>Milk chocolate</td>
</tr>
<tr>
<td>Robot for medical treatment</td>
<td>Nursing-care robot</td>
</tr>
<tr>
<td>Philosophy</td>
<td>Philosophy of language</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table V</th>
<th>EXAMPLE OF INCORRECT IS-A RELATIONSHIPS BY SCRAPING OF THE TABLE OF CONTENTS HEADINGS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Root Class</td>
<td>Is-a Hierarchy to Reef</td>
</tr>
<tr>
<td>War</td>
<td>War - Allowance reason for right of self-defense</td>
</tr>
<tr>
<td>Encryption</td>
<td>Reference: Example of code - Example 1</td>
</tr>
<tr>
<td>Piano</td>
<td>Piano - Other types</td>
</tr>
<tr>
<td>Library</td>
<td>The National Library of Japan</td>
</tr>
<tr>
<td></td>
<td>- The National Diet Library of Japan</td>
</tr>
<tr>
<td>Tokyo</td>
<td>Tokyo - Economy</td>
</tr>
<tr>
<td></td>
<td>- Information infrastructure</td>
</tr>
</tbody>
</table>
by the infobox templates through the modeling of 40 types of infobox templates.

As a result of comparing the number of property types of the infobox templates and the number of property types extracted from the articles, the average number of property types held by the infobox templates was 29.7, while the average number extracted from the articles was 48.9. One can see that the number of property types held by infoboxes extracted from articles is 1.65 times larger than the property types defined by the infobox templates.

E. Extraction Results and Observations of Synonyms

We extracted 313,527 redirect links, 24,733 class synonyms, and 81,938 instance synonyms. The accuracy rate was 67.0 ± 2.90%. Table IX shows examples of extracted synonyms and Table X shows incorrect part of synonyms we extracted. You can see that we were not able to accurately extract synonyms for class and instance directly from redirect links. However, since we used only the minimal scraping rule, we believe the accuracy of synonym extraction can be raised by supplementing the scraping rule.

V. EVALUATION AND OBSERVATIONS OF THE OVERALL WIKIPEDIA ONTOLOGY

A. Overall View and Characteristics of the Wikipedia Ontology

Table XI shows the number of classes, properties and instances of the Wikipedia Ontology. The is-a relationships, class-instance relationships, infobox triples, property domains, number of synonym relationships, and the 95% accuracy confidence interval of the Wikipedia Ontology are shown in Table XII. The total number of relationships is approximately 1.7 million, indicating that a very large-scale ontology was built through ontology learning.

We further compared the Wikipedia Ontology to Japanese WordNet and EDR [8], which are representative, preexisting, general-purpose ontologies. We then evaluated them. Table XIII shows the result. In the column of 'Ont' in Table XIII, ‘Wiki’ means Wikipedia Ontology, ‘W’ means WordNet, and ‘E’ means EDR. As in the classes “jazz guitarists” and “British air-to-air missile,” a characteristic of the Wikipedia Ontology is the fact that it defines a detailed class hierarchy specialized in a specific field. Further, the classes of the Wikipedia Ontology have a huge number of instances. These are characteristics of the Wikipedia Ontology that do not exist in other existing ontologies.

We next examined the number of classes that were roots of the class hierarchy and the paths of all leaf classes from each root. The number of all root classes was 7,211, the number of leafs was 65,721, and the number of extracted paths was 257,313. The average depth of the hierarchies of the entire structure was approximately 5.83. Finally, we measured the distribution of leafs for each root class to see across the entire ontology. The root class is on the horizontal axis and the depth of the class hierarchy is on the vertical axis of Figure 6.

From Figure 6, we observed that a lot of trees spread out, with no consolidation in one tree. Although there are

by the infobox templates through the modeling of 40 types of infobox templates.

As a result of comparing the number of property types of the infobox templates and the number of property types extracted from the articles, the average number of property types held by the infobox templates was 29.7, while the average number extracted from the articles was 48.9. One can see that the number of property types held by infoboxes extracted from articles is 1.65 times larger than the property types defined by the infobox templates.

E. Extraction Results and Observations of Synonyms

We extracted 313,527 redirect links, 24,733 class synonyms, and 81,938 instance synonyms. The accuracy rate was 67.0 ± 2.90%. Table IX shows examples of extracted synonyms and Table X shows incorrect part of synonyms we extracted. You can see that we were not able to accurately extract synonyms for class and instance directly from redirect links. However, since we used only the minimal scraping rule, we believe the accuracy of synonym extraction can be raised by supplementing the scraping rule.

V. EVALUATION AND OBSERVATIONS OF THE OVERALL WIKIPEDIA ONTOLOGY

A. Overall View and Characteristics of the Wikipedia Ontology

Table XI shows the number of classes, properties and instances of the Wikipedia Ontology. The is-a relationships, class-instance relationships, infobox triples, property domains, number of synonym relationships, and the 95% accuracy confidence interval of the Wikipedia Ontology are shown in Table XII. The total number of relationships is approximately 1.7 million, indicating that a very large-scale ontology was built through ontology learning.

We further compared the Wikipedia Ontology to Japanese WordNet and EDR [8], which are representative, preexisting, general-purpose ontologies. We then evaluated them. Table XIII shows the result. In the column of 'Ont' in Table XIII, ‘Wiki’ means Wikipedia Ontology, ‘W’ means WordNet, and ‘E’ means EDR. As in the classes “jazz guitarists” and “British air-to-air missile,” a characteristic of the Wikipedia Ontology is the fact that it defines a detailed class hierarchy specialized in a specific field. Further, the classes of the Wikipedia Ontology have a huge number of instances. These are characteristics of the Wikipedia Ontology that do not exist in other existing ontologies.

We next examined the number of classes that were roots of the class hierarchy and the paths of all leaf classes from each root. The number of all root classes was 7,211, the number of leafs was 65,721, and the number of extracted paths was 257,313. The average depth of the hierarchies of the entire structure was approximately 5.83. Finally, we measured the distribution of leafs for each root class to see across the entire ontology. The root class is on the horizontal axis and the depth of the class hierarchy is on the vertical axis of Figure 6.

From Figure 6, we observed that a lot of trees spread out, with no consolidation in one tree. Although there are
some trees with especially deep hierarchies, these are trees in which the root is a main Wikipedia category. We see from these that the upper level and intermediate level concepts are inadequate.

VI. WiLD: WIKIPEDIA LINKED DATA APPLICATION

There is currently a lot of published, mutually linked RDF data in existence on the Web. In particular, there is a strong tendency toward publishing and sharing RDF (linked data) databases that describe instances that are concrete objects.

Thus, we introduce WiLD (Wikipedia Linked Data Application), a search support tool that uses linked data implemented for use with the Wikipedia Ontology. Implementation is chiefly divided into the Wikipedia Ontology search API module and the search interface module, each implementing the Java Servlet and Adobe Flex, respectively, as a base. The WiLD system architecture is shown in Figure 7.

The API outputs the concepts related to the Wikipedia Ontology in RDF/XML format in response to key words entered by the user through the search interface. The search interface module, which accepts the output RDF/XML, edits the data into a format that is easy to read by the user and displays it in the browser. The user views the output results and is also able to send other inputs to the search API. Through this computer interaction, the user is able to acquire the desired search results. Figure 8 shows a screenshot of WiLD.

Main functions of WiLD system are shown as below.

- Wikipedia Ontology Search Functions
  - A function that displays class-instance relationships related to the search class as a graph.
  - A function that displays a tree of classes belonging to concepts searched from root classes.
  - A function that displays the triple data of a searched instance.
  - A function that displays a list of instances from classes that were selected.
  - A function that displays the attributes possessed by a class.
- functions for searching external databases.
  - A function that searches the Amazon database using the concepts of the Wikipedia Ontology.
Table XIII
EXAMPLE OF COMPARING THE ONTOLOGY

<table>
<thead>
<tr>
<th>Class</th>
<th>Ont Path</th>
</tr>
</thead>
<tbody>
<tr>
<td>person</td>
<td>Wiki Things - person - performer - Musician - guitarist</td>
</tr>
<tr>
<td></td>
<td>W entity - physical entity - object</td>
</tr>
<tr>
<td></td>
<td>- living thing - organism - person</td>
</tr>
<tr>
<td></td>
<td>- entertainer - performer</td>
</tr>
<tr>
<td></td>
<td>- musician - guitarist</td>
</tr>
<tr>
<td></td>
<td>E concept - things - object - concrete thing</td>
</tr>
<tr>
<td></td>
<td>- person who limits it in occupation, title, and role</td>
</tr>
<tr>
<td></td>
<td>- person who catches in role</td>
</tr>
<tr>
<td></td>
<td>- person who catches by occupation</td>
</tr>
<tr>
<td></td>
<td>- person who is entering art</td>
</tr>
<tr>
<td></td>
<td>- person who catches in role in music</td>
</tr>
<tr>
<td></td>
<td>- singer - musician</td>
</tr>
<tr>
<td>inanimate object</td>
<td>Wiki culture and history - Event - Politics - administration - Military affairs - weapon - British air-to-air missile</td>
</tr>
<tr>
<td></td>
<td>W entity - physical entity - object - whole</td>
</tr>
<tr>
<td></td>
<td>- artifact - instrumentality</td>
</tr>
<tr>
<td></td>
<td>- device - instrument - weapon</td>
</tr>
<tr>
<td></td>
<td>- missile - air-to-air missile</td>
</tr>
<tr>
<td></td>
<td>E concept - things - object - concrete thing</td>
</tr>
<tr>
<td></td>
<td>- concrete thing - still life</td>
</tr>
<tr>
<td></td>
<td>- concrete thing caught by function</td>
</tr>
<tr>
<td></td>
<td>- apparatus - tool used to war</td>
</tr>
<tr>
<td></td>
<td>- bullet weapon exploded by throwing out and dropping it</td>
</tr>
<tr>
<td></td>
<td>- flight weapon named missile</td>
</tr>
<tr>
<td>Abstraction thing</td>
<td>Wiki &quot;past&quot; is undefined</td>
</tr>
<tr>
<td></td>
<td>W entity - abstract entity</td>
</tr>
<tr>
<td></td>
<td>- abstraction - attribute - time - past</td>
</tr>
<tr>
<td></td>
<td>E concept - time - time point - period - present,past,future - past</td>
</tr>
</tbody>
</table>

- A function that searches a restaurant database using the concepts of the Wikipedia Ontology.
- Functions that compares and analyzes data from an external database.
- A function that compares and analyzes attribute values between concepts from infobox data.
- A function that compares and analyzes attribute values between concepts from XBRL database.

VII. CONCLUSION

In this paper, we proposed and evaluated a method of building a large-scale and general-purpose Japanese ontology through ontology learning using the Japanese Wikipedia as resources. Through this study, not only were we able to show that Wikipedia is valuable resources for ontology learning for is-a relationships, but also showed its value in extracting other relationships. Not only is it possible to extract hyponymy relationships from Wikipedia, a variety of other relationships can be extracted by focusing on them. These can be considered useful in building costless and large-scale ontologies in ontology learning.

The total numbers of articles in Wikipedia Japan as of December 2009 are over 640 thousands, which is larger compared to EDR with only 270 thousand words, and is still growing rapidly. It is predicated that the vocabulary of various fields will increase in next few years.

However, a big problem of weakness in upper ontology arose against building up higher-quality general ontology from Wikipedia. As for future work, we plan to use the Japanese WordNet as an upper level ontology and extract upper level concepts through ontology matching. In addition, we will investigate the expansion of ontology size and refinement of its properties.

In the future, we would like to provide the Wikipedia Ontology in our project page [11].

REFERENCES


286