# Constructing a Class Hierarchy with Properties by Refining and Aligning Japanese Wikipedia Ontology and Japanese WordNet

Takeshi Morita[1], Susumu Tamagawa[2], and Takahira Yamaguchi[2]

[1] School of Social Informatics, Aoyama Gakuin University, Japan
t_morita@si.aoyama.ac.jp
[2] Graduate School of Science and Technology, Keio University, Japan

**Introduction** We have proposed learning methods for building a large-scale and high accuracy general ontology called Japanese Wikipedia Ontology (JWO) by extracting the concepts and relationships between concepts from various semi-structured resources in Japanese Wikipedia [3]. However, JWO has problems because it lacks upper classes and appropriate definitions of properties. Thus, the aim of our research was to complement the upper classes in JWO by aligning JWO and Japanese WordNet (JWN) [3] using ontology alignment(OA) techniques. To achieve our aim, we developed tools that help users to refine class-instance relationships, to identify the JWO classes that need to be aligned with JWN synsets, and to align the JWO classes with the JWN synsets via user interaction. We also integrated JWO and JWN by using a domain ontology development environment, DODDLE-OWL [1]. Moreover, we propose a method for building a class hierarchy with defined properties by elevating common properties defined in sibling classes to higher classes in JWO. This research is based on our previous study [2]. The refined JWO and source code of the developed tools can be downloaded via a GitHub repository [4].

**Proposed Methods** We propose two main types of method: aligning JWO and JWN; and defining the domains of properties based on a consideration of property inheritance. Note that we used the version of JWO from November 2010 and JWN ver. 1.1 in this study. The details of the proposed methods are described in [2]. The procedures used for aligning JWO and JWN are described as follows:

1. Extracting class-instance relationships from the listing pages of Japanese Wikipedia
2. Refining class-instance relationships and identifying alignment target classes
3. Aligning JWO classes and JWN synsets
4. Integrating JWO and JWN using DODDLE-OWL
5. Removing redundant class-instance relationships

[3] http://nlpwww.nict.go.jp/wn-ja/index.en.html
[4] http://t-morita.github.io/JWO_Refinement_Tools/

We used OA techniques to integrate JWO and JWN. OA is usually applied to similar structured domain ontologies. However, the structure of JWO is quite different from that of JWN. Therefore, it is difficult to apply OA techniques using glosses, common instances or properties, and the class hierarchy structure in the two ontologies. Thus, we used methods based on string matching similarity (prefix, suffix, edit distance, and n-gram) as OA techniques to integrate JWO and JWN. The methods we selected are very basic OA techniques and the accuracy of the alignments may be low. Therefore, we developed a tool that supports the alignment of classes in JWO and the synsets in JWN via user interaction. The inputs for the tool are the alignment target classes in JWO. A user can dynamically align the classes in JWO and the synsets in JWN. The user aligned 736 alignment target classes in JWO and sysnsets in JWN using the tool in about 6 hours.

As a result, the number of classes from JWO is 2,787, the number of classes from JWN is 675, the number of instances is 344,934, and the number of class-instance relationships is 444,597.

The procedure of defining domains of properties based on a consideration of property inheritance by refining the definition of the domains of properties in JWO is as follows:

1. Extracting the domains of properties from instance triples and the types of subject resources for the instance triples (If there is an instance triple s-p-o and the type of s is T, the domain of property p is T. )
2. Elevating common properties that are defined in the sibling classes to higher classes in JWO
3. Removing the properties defined in a class that are also defined in superclasses of the class (The properites can be derived using a reasoner, so we regard them as redundant properties and remove them.)

As a result, we extracted 4,357 properties. After elevating the properties and removing the redundant properties that are defined expressly, we reduced the number of domains of properties from 143,500 to 18,678.

## References

1. Takeshi Morita, Noriaki Izumi, Naoki Fukuta, Takahira Yamaguchi, *DODDLE-OWL: Interactive Domain Ontology Development with Open Source Software in Java*, IEICE Transactions on Information and Systems, Special Issue on Knowledge-Based Software Engineering Vol.E91-D No. 4 pp. 945-958, 2008.
2. Takeshi Morita, Yuka Sekimoto, Susumu Tamagawa and Takahira Yamaguchi, *Building up a class hierarchy with properties by refining and integrating Japanese Wikipedia Ontology and Japanese WordNet*, Web Intelligence and Agent Systems: An International Journal, Volume 12, Number 2, pp. 211-233, IOS Press, 2014.
3. Susumu Tamagawa, Shinya Sakurai, Takuya Tejima, Takeshi Morita, Noriaki Izumi, and Takahira Yamaguchi, *Learning a Large Scale of Ontology from Japanese Wikipedia.* 2010 IEEE/WIC/ACM International Conference on Web Intelligence, pp. 279-286, 2010.