

Automated Ontology Framework for Service Robots

Saranya Kanjaruek, Dayou Li, Renxi Qiu and Noppakun Boonsim

Abstract— This paper presents an automated ontology framework for service robots. The framework is designed to automatically create an ontology and an instance of concept in dynamic environment. Ontology learning from text is applied to build a concept hierarchy using WordNet which provides a rich semantic processing for physical objects. The Automated Ontology is composed of four modules: Concept Creation, Property Creation, Relationship Creation and Instance of Concept Creation. The automated ontology algorithm was implemented in order to create the concept hierarchy in the Robot Ontology. The Semantic Knowledge Acquisition represents knowledge of physical objects in dynamic environments. In simulation experiments, the list of object names and property names was identified. The result shows the concept hierarchy which represents explicit terms and the semantic knowledge of physical objects for performing everyday manipulation tasks.

I. INTRODUCTION

An ontology provides explicit knowledge and shares concepts between humans and household robots in order for the robots to understand objects in the environment. Currently, a service robot ontology is often manually encoded before the robots are deployed. This leads to the situation where the robots cannot understand objects that do not exist in the predefined ontology in a dynamic environment. Domestic environments are dynamic. A dynamic ontology is expected to be created by the robots themselves when they perceive objects. Thus, the challenge is how robots understand objects in dynamic environments and create an ontology automatically. If the robots cannot understand the objects, then they would not be able to allocate objects when serving their human users in domestic environments.

This paper presents a systematic architecture and an algorithm for domestic robots to automatically create and manipulate ontology in domestic environments. The architecture contains vocabulary, taxonomy and a set of rules for the robots to perceive physical objects in domestic environments they share with their human users. The algorithm enables the robot to perform ontology learning

Saranya Kanjaruek is PhD student in the Institute for Research in Applicable Computing, University of Bedfordshire, UK.

Dayou Li is Professor in the Institute for Research in Applicable Computing, University of Bedfordshire, UK.

Renxi Qiu is a Senior Research Fellow and Robotic Lab Manager in the Institute for Research in Applicable Computing, University of Bedfordshire, UK.

Noppakun Boonsim is PhD student in the Institute for Research in Applicable Computing, University of Bedfordshire, UK.

from text and then create the ontology. WordNet is used for identifying objects and retrieving semantic relations of physical objects.

The rest of the paper is organized as follows. Section II presents related works. Section III describes the Data-Information Retrieval based Automated Ontology Framework. Section IV proposes the Automatic Ontology process. Section V and Section VI present the experiment and conclusion, respectively.

II. RELATED WORKS

An ontology is a set of formal representation terms with explicit specifications of a shared conceptualization [1]. An ontology is presented as a vocabulary, a taxonomy and a set of rules in the form of precise semantics. The use of an ontology as a basis for designing and modelling robots is developed to formally describe all objects and the relationships between them. Open Robot Ontology [2] (ORO) is designed as the central knowledge storage service which can be represented by RDF, OWL2 and SROIQ. The ORO server identifies a new agent and automatically creates a new in-memory OWL model for the agent. KnowRob [3] is the robot knowledge processing framework which includes robot control programs and allows reasoning methods which connect to the robot's perception and action systems.

Ontology learning from text [4] is an automatic or a semi-automatic process of ontology construction and maintenance, identifying terms, concepts, relations, and axioms from textual information. There are five types of output in ontology learning: terms, concepts, taxonomic relations, non-taxonomic relations and axioms. Terms are defined as lexical realization which are single word or multi word and relevant to a domain. Concepts are created by grouping similar terms. A taxonomic relation organizes concepts into a hierarchy. Non-taxonomic relations are the interactions between the concepts. Axioms are sentences that examine and define the correctness of ontology. The techniques can be classified into statistics-based, linguistics-based and logic-based. The statistics-based techniques are derived from information retrieval, machine learning and data mining. The syntagmatic similarity is the association between terms. There are two types of syntagmatic similarity, namely, semantic similarity and semantic relatedness. The concept of semantic similarity is more specific than semantic relatedness and measures the degree to which two concepts are similar. Two concepts are connected through hierarchical is-a relations. The linguistics-based techniques are dependent on natural language processing. Some of the techniques include semantic lexicons, lexico-syntactic and part-of-speech tagging. The semantic lexicons implement a large collection

of predefined concepts and relations, such as WordNet. There are two logic-based techniques, inductive logic programming and logical inference. The inductive logic programming rules are obtained from existing collection of concepts and relations.

WordNet [5] is a large lexical database of English. Terms are assigned senses by synsets. A synset is a collection of terms which share the same meaning. Nouns, verbs, adverbs and adjectives are organized into synsets. There are five commonly used semantic relations for nouns: synonym, hyponym, hypernym, holonym and meronym. Synonym is a word or phrase that means exactly or nearly the same as another word or phrase. Hypernym is a word with a broad meaning. Hyponym is a word of more specific meaning than hypernym. Holonym is used for naming the whole of which a given word is a part. Meronym is used for naming a part of a larger whole. Hierarchy creation is the method to create a hierarchy. There are various methods available. The tree ascending algorithm [6] is combined the distributional similarity measures and the taxonomic similarity between nearest neighbors. The distributional similarity measures the similarity of two concepts in text corpora. The taxonomic similarity measures the similarity of concepts in a taxonomy structure.

LexOnt [7] creates a OWL ontology of service classes and generates a top-n list of terms. Then, a user chooses keywords to assign as features and enters these terms into the ontology processing. It creates a OWL instance and adds a hasFeature property in order to construct the ontology to generate a new list of top-n terms. The ontology is built by taking in the terms that the user has chosen. The robot technology ontology [8] consists of a Where layer, a What layer and a How layer based on 4W1H and three classes of semantic concepts. This structure extracts objects and human activities from education books and MIT's ConceptNet. This work is based on the extraction by considering the grammatical relation between verb and noun in sentences.

The InPhO [9] project is a dynamic ontology in which the taxonomic structures are generated by machine reasoning. The expert feedback module extracted relationships from the *Stanford Encyclopedia of Philosophy*. This ontology is generated by a combination of automatic and semi-automatic methods.

The DYNAMO [10] ontology for information retrieval addresses the improvement of semantic information retrieval driven by users' satisfaction. There are three corpora which serve as knowledge source and two ontology approaches are performed in this project.

KnowLang [11] is a formal language with knowledge representation and reasoning for modelling automatic self-adaptive behaviour and is awareness-based. It provides an explicit representation of domain concepts and relationships, the explicit representation of particular and general factual knowledge and also uncertain knowledge. The next section describes the Data-Information Retrieval based Automated Ontology Framework

I. DATA-INFORMATION RETRIEVAL BASED AUTOMATED ONTOLOGY FRAMEWORK

The Data-Information Retrieval based Automated Ontology Framework consists of two phases and seven processes as shown in Fig.1.

The first phase is the automated ontology. All objects are recognized in the Data Input process as text. The Automatic Ontology process executes the list text file, classifying objects and properties, creating the relationships and creating the instances. The unknown objects are transferred to the Data Retrieval and Information Retrieval process and after this process has executed, it returns the name of each concept as text to the Automatic Ontology process. The concepts and properties are created as an ontology hierarchy into the Robot Ontology. The Semantic Knowledge Acquisition represents the knowledge of objects in the form of an instance of concept in the dynamic environment.

The second phase is the use of the automated ontology from the user point of view. The User Query process queries the Robot Ontology, presents the query results as text to the user and sends the query results to the Result Evaluation. The Result Evaluation process retrieves the result from the User Query process in order to assess the result. The following describes each process in detail.

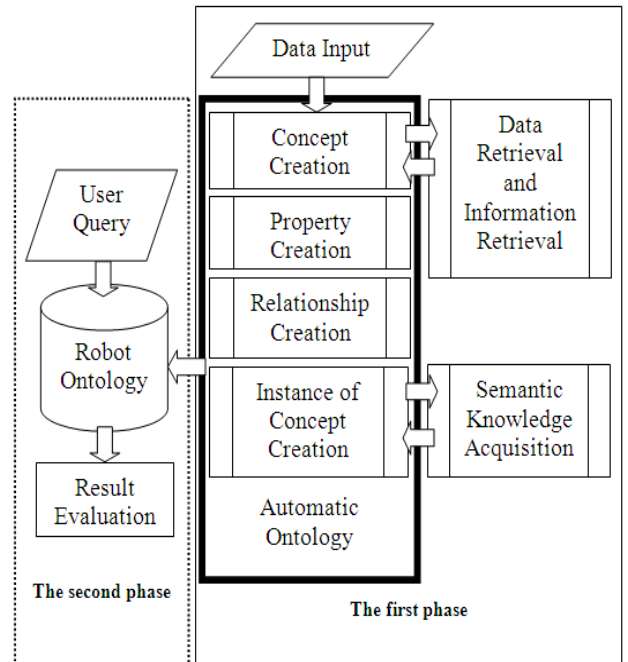


Figure 1. Data-Information Retrieval based Automated Ontology Framework

A. Data Input process

Object recognition is a very active area of research. This framework assumes that all objects have been recognized in the Data Input process. The list of object names and property names is acquired in order to classify this data in the Automatic Ontology process.

B. Automatic Ontology process

The Automatic Ontology is proposed in order to manage and update the Robot Ontology. There are four modules in this process: (1) Concept Creation, (2) Property Creation, (3) Relationship Creation and (4) Instance of Concept Creation. This process is designed not only to create the Automatic Ontology but also to tackle unknown objects which do not exist in the WordNet by requiring data from the Data Retrieval and Information Retrieval process. Section IV explains the Automatic Ontology.

C. Data Retrieval and Information Retrieval process

The Data Retrieval and Information Retrieval process is designed to support unknown objects which do not exist in the WordNet by receiving data of unknown objects from the Automatic Ontology process. This process combines Data Retrieval and Information Retrieval and combines them in order to retrieve a fixed set of data-base and natural language documents. The search results are sent to the Automatic Ontology process.

D. Semantic Knowledge Acquisition process

The Semantic Knowledge Acquisition solves the problem of dynamic ontology. It consists of four sub-modules. Firstly, the FileN Algorithm module compares the list of objects from the Data Input process with the Robot Ontology process. Secondly, the Semantic Representation module represents all objects and properties in the Robot Ontology. Thirdly, the Ontology Updating module accesses and manipulates the Robot Ontology. Finally, the Object Prediction module determines the future location of a specific object.

E. Robot Ontology process

The Robot Ontology contains the dynamic ontology for the robot represented in OWL. A OWL ontology presents common concepts and sharable knowledge using class (concept), properties and individuals (instance). This ontology is not only adaptable for providing knowledge to robots in order to answer a query from the User Query process but can also be automatically manipulated by the Automatic Ontology process.

The Robot Ontology is designed with two levels, as shown in Fig.2: the Ontology Level and the Semantic Knowledge Level, in order to competently perform everyday tasks. The Ontology Level presents a concept hierarchy and relationships between concepts. The Semantic Level presents instances of concept and properties.

F. User Query process

The User Query module receives a query from a user and presents the result from the Robot Ontology. The query is transformed into the query language in order to process it in the Robot Ontology. In the Robot Ontology, the hierarchy was created in order to facilitate and provide an answer to the user. There are two types of User Query process: the location prediction query for a specific object and the similar object query for a specific object. The location prediction query is

implemented in the Semantic Knowledge Acquisition. The similar object query is addressed by the hierarchy in the Robot Ontology. Synonyms of the concept present similar objects. When a robot operates on an everyday task, it should present similar objects within the same category to the user query.

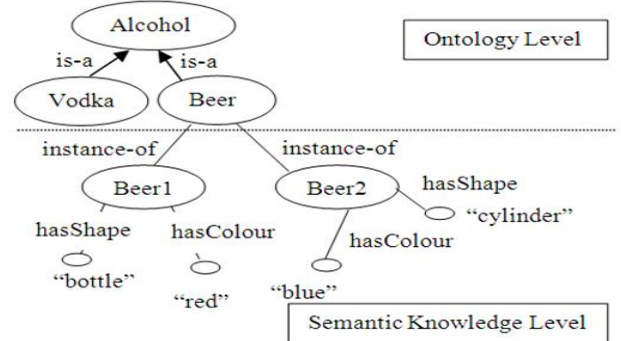


Figure 2. The Robot Ontology structure

G. Result Evaluation process

The Result Evaluation module aims to validate the results from the Robot Ontology before providing results to the robot and the user.

II. AUTOMATIC ONTOLOGY

The aim of this work is to manipulate an ontology without human involvement in dynamic environments. The Automatic Ontology process is a proposed automated ontology-building process which consists of four modules as described in the following.

A. Concept Creation module

Identifying concepts and constructing a hierarchy are the main objectives of this module. The list of object names and property names are acquired from the Data Input process. The object names are identified by using the concepts and glossary from WordNet by searching for the synonym(s) and hypernym(s) of a physical object. The concepts and taxonomic relations are discovered using hypernym(s) from the WordNet to construct the concepts into a concept hierarchy. The Concept Creation module starts by checking an object name in the Robot Ontology. If an object name exists in the Robot Ontology then it creates an instance of concept and send the instance and properties of instance to the Instance of Concept Creation module. The first step in the automated ontology algorithm, as shown in Fig.3, is to get the synset of an object name from the WordNet and to search in the Robot Ontology. If the synonym of an object name is found in the Robot Ontology, this object name is created as a concept and a sibling relationship between an object name and synonym is established. This means that an object name and its synonyms have same parent concept.

On the other hand, if this object name is created as a new concept, the hypernym of this concept is utilized in order to create a concept hierarchy. The concept hierarchy is constructed from the bottom-up processing and an is-a

relationship is created between parent and child concepts. The parent concept of an object name concept is set as level 0 and the first hypernym of the new concept is set as level 1. The level increases by 1 until the physical_entity level which indicates this is a physical object for the robot's perception. Before creating a concept in each hypernym level, whether the concept exists in the Robot Ontology must be checked in order to avoid duplicate concepts. The object name which does not exist in the WordNet is sent to the Data Retrieval and Information Retrieval process.

```

Input: object_name
Get the synset of an object name
If object_name and synset not in Robot Ontology
Then
  If object_name in WordNet
  Then
    Create object_name concept
    For I = 0
      Get from hypernym level I to physical_entity level
      I=I+1
    Exit For
    Candidate concepts = level 1 to physical_entity level
    Level 0= {cJ, cJ+1...}, J j=1 to n

    If level 0 not in Robot Ontology
    Then //find the parent of object_name concept

      For parent (level 0) J =1 to n

        If synonym (level 0) is physical_entity
        or hyponym (level 0) is physical_entity Then

          If Parent (level 0) in Candidate concepts then
          Is-a (object_name concept, level 0 concept)
          Else if parent (level 0) in Candidate concepts and
          parent (level 0) is physical_entity
            Is-a (level 0 concept, physical_entity)
          Else if parent (level 0) in Candidate concepts and
          synonym (level 0) is Physical_entity
            Is-a (level 0 concept, physical_entity)

          Set New Level 0= {cJ, cJ+1...}, J =1 to n

          Else
            J=J+1
          Exit For

        Else //level 0 in RO
        Is-a (object_name concept, level 0 concept)

      Else //object_name not in WordNet
      Active DIR

    Else if synset in Robot Ontology
    Is-a (object_name concept, parent (synset))

  Else
    Create instance of object_name concept
  
```

Figure 3. The automated ontology algorithm

B. Property Creation module

The Property Creation module aims to create new properties and send to the Semantic Knowledge Acquisition process. Moreover, the rule base is applied to identify properties of the concept, as shown in Fig.4. The list of physical properties of objects is acquired from the Data Input process. The Rule base is implemented to classify the appearance properties of objects into seven predefined object properties: hasColour, hasShape, hasMaterial, hasBrand, hasPart, hasLocation and hasProperty, which are relationships between two instances.

<p>Rule 1 IF Pvalue is colour name</p> <p>THEN Dproperty has hasColour property and Pvalue = colour name</p>
<p>Rule 2 IF Pvalue is cylinder or square or flat or dome or sharp or circle or oval or triangle or hexagon or cuboids or box or vase or cone or cube or sphere or pyramid or rectangular prism</p> <p>THEN Dproperty has hasShape property and Pvalue = shape name</p>
<p>Rule 3 IF Pvalue is handle or leg or arm or cover or lid or wheel or fur or button or wire or bottle cap or bottle top or screw cap or cork stopper or sport cap or bottle closer or pull-off bottle cap</p> <p>THEN Dproperty has hasPart property and Pvalue = part name</p>
<p>Rule 4 IF Pvalue is wood or plastic or cloth or paper or cement or leather or glass or cotton or silk or plywood or gold or silver or platinum or stone or copper or tin or hardness or elasticity or tensile strength or brittleness or adhesiveness</p> <p>THEN Dproperty has hasMaterial property and Pvalue = material name</p>
<p>Rule 5 IF Pvalue is a proper noun</p> <p>THEN Dproperty has hasBrand property and Pvalue = brand name</p>
<p>Rule 6 IF Pvalue is a location</p> <p>THEN Dproperty has hasLocation property and Pvalue = location</p>
<p>Rule 7 ELSE Dproperty has hasProperty property and Pvalue = property name</p>

Figure 4. The Rule for Property Creation module

C. Relationship Creation module

The Relationship Creation module contains the method for how new objects are related to other objects in the Robot Ontology. The taxonomic relations represent relations which organize concepts into hierarchies. There are two types of taxonomic relation in this framework, is-a relations, and sibling relations. The new concept and its hypernym are created. The is-a relation is utilized in order to create relationships between concepts. The Semantic Knowledge Acquisition represents the object properties and the data properties. The object properties are relationships between two instances. The data properties link an instance to a XML schema datatype value or a RDF literal.

D. Instance of Concept Creation module

The Instance of Concept Creation module is designed to manipulate and define the instance of concept which receives data from the Concept and Property Creation module. The instance of concept is created and is sent to the Semantic Knowledge Acquisition in order to represent the properties of instance.

III. EXPERIMENT AND ANALYSIS

The experiment shows the new object is created in the Robot Ontology. The Data Input is a list text file as shown in Fig.5. The object name (medicine) is identified by using the concepts and glossary from the WordNet in the Concept Creation module. This experiment assumes that the Robot Ontology is empty. The synonyms of medicine are medical_science, learned_profession, drug and punishment. Neither the medicine concept nor synonyms exist in the Robot Ontology. As a consequence, the medicine concept is created. The hypernym of medicine consist of three levels from level 0 to level 2 as shown in Fig.6. All the concepts from level 1 to level 2 are assigned as candidate concepts. The concept hierarchy is only selected from the candidate concepts.

```

medicine,[1,1,1.5,-1]
Pepsi,[1,1,0.5,-1]
beer,[1,-1,1,1]
cocktail,[1,-1,0.5,1]
    
```

Figure 5. The list text file

```

Level0 is Parent of medicine = {agent, profession, social_control}
Level1 = {group_action, causal_agent, natural_science, occupation,
substance}
Level2 = {event, science, matter, act, physical_entity}
    
```

Figure 6. The hypernym of the medicine concept

Firstly, the parent of the medicine concept is selected from the agent concept or/and the profession concept or/and the social_control concept. In order to select the parent of the medicine concept, at least one member of parent of the agent concept or/and the profession concept or/and the social_control concept appears in the candidate concepts as shown in Fig.7. The selected concept is the agent concept because the matter concept is in the candidate concepts (is-a (medicine, agent)) Moreover, the parent of agent is physical_entity (is-a (agent, physical_entity)). However, social_control is not selected because parent of social_control or hyponym of social_control is not physical_entity.

```

Parent of agent = {matter, capitalist, negotiator, skilled_worker,
physical_entity}
Parent of profession = {commitment, activity, body, assertion}
Parent of social_control = {act, event}
    
```

Figure 7. The parent concepts for the agent, profession and social_control concept

There is one property of medicine, as shown in Fig.5. The Property Creation module creates the property of medicine concept as shown in Fig.8. The DataProperty of medicine is hasLocation and Literal is [1, 1, 1.5, -1] value. The Instance of Concept Creation module creates the instance of medicine concept in order to present the physical object of medicine in the environment after the medicine concept is created. The unique number is assigned to the first instance of the medicine concept. It starts at 1 and increments by 1.

```

<ClassAssertion>
  <Class IRI="#medicine"/>
  <NamedIndividual IRI="#medicine1"/>
<ClassAssertion>
<Declaration>
  <NamedIndividual IRI="#medicine1"/>
<Declaration>
<DataPropertyAssertion>
  <DataProperty IRI="#hasLocation"/>
  <NamedIndividual IRI="#medicine1"/>
  <Literal datatypeIRI="#rdf:PlainLiteral"> [1, 1, 1.5, -1] </Literal>
<DataPropertyAssertion>
    
```

Figure 8. The instance of the medicine concept

```

<Declaration>
  <Class IRI="#SRP"/>
<Declaration>
<SubClassOf>
  <Class IRI="#Instan"/>
  <Class IRI="#SRP"/>
</SubClassOf>
<Instan IRI="#medicine1">
  <AreaAppear CountOccurrence="1"> [1, 1, 1.5, -1] </AreaAppear>
</Instan>
    
```

Figure 9. The instance of medicine concept in the Semantic Knowledge Acquisition

The instance is sent to the Semantic Knowledge Acquisition in order to represent the properties of instance, as shown in Fig.9 in OWL language and this semantic knowledge is created by xml [12]. The hasLocation value of the instance of medicine is the AreaAppear concept value. The hierarchy of the Semantic Knowledge Acquisition is made of OWL language. The Instan concept is a subclass of the SRP concept. The Instan concept has an IRI property which stores the value of the instance name. The AreaAppear concept is the child concept of the Instan concept. The AreaAppear concept has CountOccurrence property and it stores the location value of the instance. The CountOccurrence property shows the number of occurrences of the instance. It counts the number of occurrences of this instance which located at this location in order to compute the probability of prediction the location of this instance.

Next, the Pepsi object name, the beer object name and the cocktail object name are identified in WordNet by following the automated ontology algorithm. The concept hierarchy is shown in Fig.10. The concept hierarchy consists of 13 concepts. All instances are created in the Semantic Knowledge Acquisition in order to represent the properties of the instance, as shown in Fig.11.

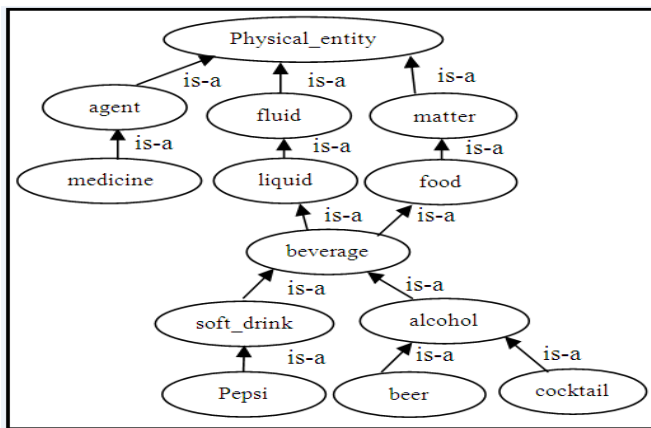


Figure 10. The concept hierarchy

```

<Instan IRI="#medicine1">
  <AreaAppear CountOccurrence="1"> [1, 1, 1.5, -1] </AreaAppear>
</Instan>
<Instan IRI="#Pepsi1">
  <AreaAppear CountOccurrence="1"> [1, 1, 0.5, -1] </AreaAppear>
</Instan>
<Instan IRI="#beer1">
  <AreaAppear CountOccurrence="1"> [1,-1, 1, 1] </AreaAppear>
</Instan>
<Instan IRI="#cocktail1">
  <AreaAppear CountOccurrence="1"> [1,-1, 0.5, 1] </AreaAppear>
</Instan>
  
```

Figure 11. The instances of concept in the Semantic Knowledge Acquisition

IV. CONCLUSION

This paper presents the Automated Ontology Framework for Service Robots automatic ontology creation for household robots. The Automated Ontology enables the robots to understand objects and the Semantic Knowledge Acquisition represents the knowledge in a way that the robots can easily use and manipulate. The initial simulation experiments have shown the feasibility of building the concept hierarchy by using the automated ontology algorithm with WordNet and creating instances of concept in the Semantic Knowledge Acquisition.

Future work will focus on the Result Evaluation and the Data Retrieval and Information Retrieval processes for completing the framework.

ACKNOWLEDGMENT

This work is supported with a PhD research scholarship provided by Khon Kaen University, Thailand.

REFERENCES

- [1] T. R. Gruber, "A translation approach to portable ontology specifications", *Knowledge Acquisition*, 1993, 5 (2), pp.199-220.
- [2] S. Lemaignan, M. Ros, R.L. Alami, and M. Beetz, Oro, a knowledge management module for cognitive architectures in robotics, in *Proc. 2010 IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems*, Taipei, 2010, pp.3548-3553.
- [3] M. Tenorth and M. Beetz, "KNOWROB — Knowledge processing for autonomous personal robots", *Intelligent Robots and Systems (IROS) IEEE/RSJ International Conference*, St.Louis, 2009, pp. 4261-4266.
- [4] W. Wong, W Liu, and M Bennamoun. "Ontology learning from text: A look back and into the future". *ACM Computing Surveys (CSUR)*, 2012, 44 (4) pp. 20-.
- [5] C. Fellbaum. *WordNet*. London, Blackwell Publishing Ltd, 1998.
- [6] V. Pekar, and S.Staab, "Taxonomy learning: factoring the structure of a taxonomy into a semantic classification decision", in *Proc. 19th Intl. Conf. Computational linguistics*, Stroudsburg, PA, 2002, 1, pp. 1-7.
- [7] K. Arabshian, and P. Danielsen, "Semi-automated ontology creation for high-level service classification," in *Proc. 7th Intl. Conf. Semantics Knowledge and Grid (SKG)*, Beijing, 2011 pp. 17-20.
- [8] T. L. Ngo, H. Lee, and M. Mizukawa, "Automatic building robot technology ontology based on basic-level knowledge", *Journal of Robotics and Mechatronics*, 2011, 23 (4), pp. 515.
- [9] J. Murdock, C. Buckner and C. Allen, "Evaluating dynamic ontologies", in *Anonymous Knowledge discovery, knowledge engineering and knowledge management*. Berlin, Springer, 2013, pp. 258-275.
- [10] Z. Sellami, V. Camps, and N. Aussenac-Gilles, "DYNAMO-MAS: A multi-agent system for ontology evolution from text", *Journal on Data Semantics Concepts and Ideas for Building Knowledgeable Systems*, 2013, 2 (2-3), pp.145-161.
- [11] E. Vassev, and M. Hinchey, "Knowledge-Based Self-Adaptation", in *Proc. 6th Latin-American Symp. on Dependable Computing (LADC)*, Rio de Janeiro, Brazil, April 2-5, 2013, pp. 11-18.
- [12] lxml, visited January 19, 2015. <http://lxml.de/>