Towards On-the-Fly Ontology Construction - Focusing on Ontology Quality Improvement

Naoki Sugiura[1], Yoshihiro Shigeta[1], Naoki Fukuta[1], Noriaki Izumi[2], and Takahira Yamaguchi[1]

[1] Shizuoka University, 3-5-1 Johoku, Hamamatsu, Shizuoka 432-8011, Japan,
sugiura@ks.cs.inf.shizuoka.ac.jp,
http://mmm.semanticweb.org
[2] National Institute of AIST, 2-41-6, Aomi, Koto-ku, Tokyo, Japan

Abstract. In order to realize the on-the-fly ontology construction for the Semantic Web, this paper proposes DODDLE-R, a support environment for user-centered ontology development. It consists of two main parts: pre-processing part and quality improvement part. Pre-processing part generates a prototype ontology semi-automatically, and quality improvement part supports the refinement of it interactively. As we believe that careful construction of ontologies from preliminary phase is more efficient than attempting generate ontologies full-automatically (it may cause too many modification by hand), quality improvement part plays significant role in DODDLE-R. Through interactive support for improving the quality of prototype ontology, OWL-Lite level ontology, which consists of taxonomic relationships (class - sub class relationship) and non-taxonomic relationships (defined as property), is constructed efficiently.

1 Introduction

As the scale of the Web becomes huge, it is becoming more difficult to find appropriate information on it. When a user uses a search engine, there are many Web pages or Web services which are syntactically matched with user's input words but semantically incorrect and not suitable for user's intention. In order to defeat this situation, Semantic Web[1] is now gathering attentions from researchers in wide area. Adding semantics (meta-data) to the Web contents, software agents are able to understand and even infer Web resources. To realize such paradigm, the role of ontologies[2][3] is important in terms of sharing common understanding among both people and software agents[4]. On the one hand, in knowledge engineering field ontologies have been developed for particular knowledge system mainly to reuse domain knowledge. On the other hand, for the Semantic Web, ontologies are constructed in distributed places or domain, and then mapped each other. For this purpose, it is an urgent task to realize a software environment for rapid construction of ontologies for each domain. Towards the on-the-fly ontology construction, many researches are focusing on

J. Davies et al. (Eds.): ESWS 2004, LNCS 3053, pp. 1–15, 2004.

automatic ontology construction from existing Web resources, such as dictionaries, by machine processing with concept extraction algorithms. However, even if the machine produces ontologies automatically, users still need to check the output ontology. It may be a great burden for users to check all the correctness of the ontology and modify it, especially if the scale of automatically produced ontology is large. Considering such situation, we believe that the most important aspect of the on-the-fly ontology construction is that how efficiently the user, such as domain experts, are able to check the output ontology in order to make Semantic Web contents available to the public. For this reason, ontologies should be constructed not fully automatically, but through interactive support by software environment from the early stage of ontology construction. Although it may seem to be contradiction in terms of efficiency, the total cost of ontology construction would become less than automatic construction because if the ontology is constructed with careful interaction between the system and the user, less miss-construction will be happened. It also means that high-quality ontology would be constructed. In this paper, we propose a software environment for user-centered on-the-fly ontology construction named DODDLE-R (Domain Ontology rapiD DeveLopment Environment - RDF[5] extension). The architecture of DODDLE-R is re-designed based on DODDLE-II [6], the former version of DODDLE-R. Although DODDLE-II has already provided interactive support for ontology construction, the system architecture is not well-considered and sophisticated. The DODDLE-R system is modularized into machine-processing module and user-interaction module in order to separate pre-processing part and user-centered quality management part specifically. Especially, to realize the user-centered environment, DODDLE-R dedicates to the quality improvement part. It enables us to develop ontologies with interactive indication of which part of ontology should be modified. The system supports the construction of both taxonomic relationships and non-taxonomic relationships in ontologies. Additionally, because DODDLE-II has been built for ontology construction not for the Semantic Web but for typical knowledge systems, it needs some extensions for the Semantic Web such as OWL (Web Ontology Language) [7] import and export facility. DODDLE-R supports OWL-Lite level ontology construction because if we think of user-centered ontology construction, OWL-DL or OWL-Full sounds too complicated for human to understand thoroughly. DODDLE-R contributes the evolution of ontology construction and the Semantic Web.

2 System Design of DODDLE-R

Fig. 1 shows the overview of DODDLE-R. The main feature of DODDLE-R is the modularized two parts - pre-processing part and quality improvement part. In pre-processing part, the system generates the basis of the ontology, a taxonomy and extracted concept pairs, by referring to WordNet[8] as an MRD (Machine Readable Dictionary) and domain specific text corpus. A taxonomy is a hierarchy of IS-A relationship. Concept pairs are extracted based on co-occurrency by using statistic methods. These pairs are the candidates which has

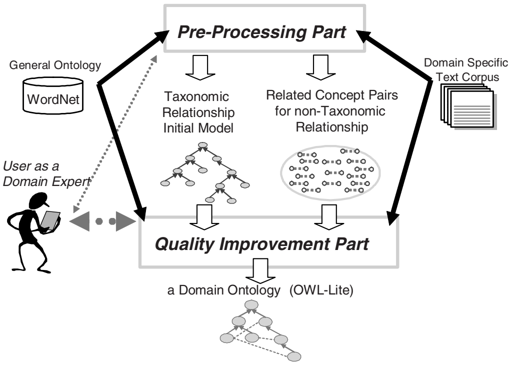

Pre-Processing Part
General Ontology
WordNet
Domain Specific Text Corpus
Taxonomic Relationship Initial Model
Related Concept Pairs for non-Taxonomic Relationship
User as a Domain Expert
Quality Improvement Part
a Domain Ontology (OWL-Lite)

Fig. 1. DODDLE-R overview

significant relationships. A user identifies some relationship between concepts in the pairs. In quality improvement part, the prototype ontology produced by pre-processing part is modified by a user through interactive support by the system.

2.1 Pre-processing Part

In pre-processing part, the system generates the basis of output ontology for further modification by a user. Fig. 2 describes the procedure of pre-processing part. This part consists of three sub-parts: input concept selection, taxonomy building, and related concept pair acquisition. First, as input of the system, several domain specific terms are selected by a user. The system shows a list of noun concepts in the domain specific text corpus as candidates of input concept. At this phase, a user also identifies the sense of terms to map those terms to concepts in WordNet.

For building taxonomic relationship (class - sub class relationship) of an ontology, the system attempts to extract "best-matched concepts". That is, "concept matching" between input concepts and WordNet concepts is done, and matched nodes are extracted, and then merged at each root nodes. To extract related concept pairs from domain specific text corpus as a basis of identifying non-taxonomic relationships (such as "part-of" relationship), statistic methods are applied. In particular, WordSpace[9] and an association rule algorithm[10] are used in this part and these methods attempt to identify significantly related concept pairs.

Construction of WordSpace WordSpace is constructed as shown in Fig.3.
1. Extraction of high-frequency 4-grams Since letter-by-letter co-occurrence information becomes too much and so often irrelevant, we take term-by-term co-occurrence information in four words (4-gram) as the primitive to make up co-

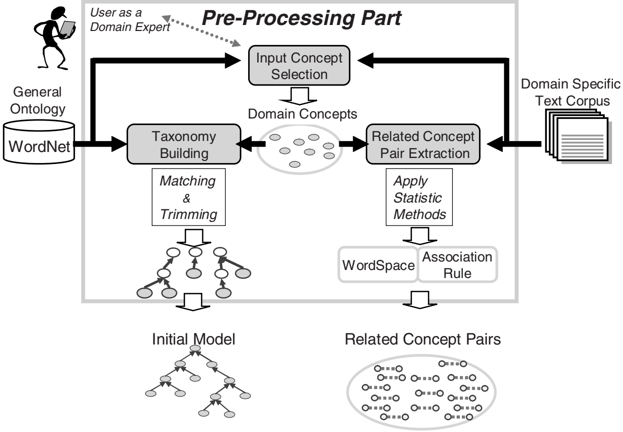


Fig. 2. Pre-processing Part

occurrence matrix useful to represent context of a text based on experimented results. We take high frequency 4-grams in order to make up WordSpace.

2. Construction of collocation matrix A *collocation matrix* is constructed in order to compare the context of two 4-grams. Element $a_{i,j}$ in this matrix is the number of 4-gram f_i which comes up just before 4-gram f_j (called *collocation area*). The collocation matrix counts how many other 4-grams come up before the target 4-gram. Each column of this matrix is the *4-gram vector* of the 4-gram f.

3. Construction of context vectors A *context vector* represents context of a word or phrase in a text. A sum of 4-gram vectors around appearance place of a word or phrase (called *context area*) is a context vector of a word or phrase in the place.

4. Construction of word vectors A word vector is a sum of context vectors at all appearance places of a word or phrase within texts, and can be expressed with Eq.1. Here, $\tau(w)$ is a vector representation of a word or phrase w, $C(w)$ is appearance places of a word or phrase w in a text, and $\varphi(f)$ is a 4-gram vector of a 4-gram f. A set of vector $\tau(w)$ is WordSpace.

$$\tau(w) = \sum_{i \in C(w)} \Big(\sum_{f \text{ close to } i} \varphi(f) \Big) \tag{1}$$

5. Construction of vector representations of all concepts The best matched "synset" of each input terms in WordNet is already specified, and a sum of the word vector contained in these synsets is set to the vector representation of a concept corresponding to a input term. The concept label is the input term.

6. Construction of a set of similar concept pairs Vector representations of all concepts are obtained by constructing WordSpace. Similarity between concepts is obtained from inner products in all the combination of these vectors. Then we

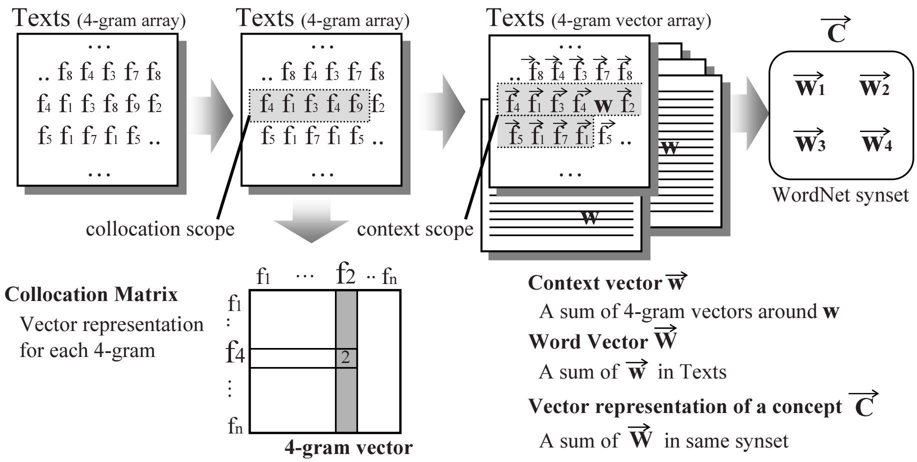


Fig. 3. Construction flow of WordSpace

define certain threshold for this similarity. A concept pair with similarity beyond the threshold is extracted as a similar concept pair.

Finding Association Rules between Input Terms The basic association rule algorithm is provided with a set of transactions, $T := \{t_i \mid i = 1..n\}$,where each transaction t_i consists of a set of items, $t_i = \{a_{i,j} \mid j = 1..m_i, a_{i,j} \in C\}$ and each item $a_{i,j}$ is form a set of concepts C. The algorithm finds association rules $X_k \Rightarrow Y_k : (X_k, Y_k \subset C, X_k \cap Y_k = \{\})$ such that measures for support and confidence exceed user-defined thresholds. Thereby, support of a rule $X_k \Rightarrow Y_k$ is the percentage of transactions that contain $X_k \cup Y_k$ as a subset (Eq.2)and confidence for the rule is defined as the percentage of transactions that Y_k is seen when X_k appears in a transaction (Eq.3).

$$support(X_k \Rightarrow Y_k) = \frac{\mid \{t_i \mid X_k \cup Y_k \subseteq t_i\} \mid}{n} \tag{2}$$

$$confidence(X_k \Rightarrow Y_k) = \frac{\mid \{t_i \mid X_k \cup Y_k \subseteq t_i\} \mid}{\mid \{t_i \mid X_k \subseteq t_i\} \mid} \tag{3}$$

As we regard input terms as items and sentences in text corpus as transactions, DODDLE-R finds associations between terms in text corpus. Based on experimented results, we define the threshold of support as 0.4% and the threshold of confidence as 80%. When an association rule between terms exceeds both thresholds, the pair of terms are extracted as candidates for non-taxonomic relationships.

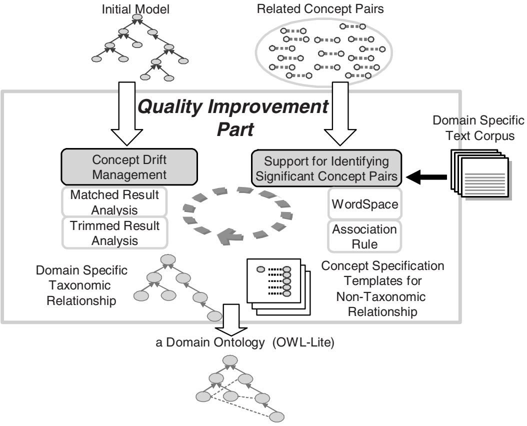
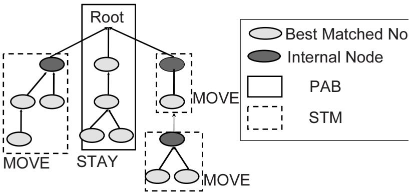

Initial Model
Related Concept Pairs
Quality Improvement Part
Domain Specific Text Corpus
Concept Drift Management
Matched Result Analysis
Trimmed Result Analysis
Support for Identifying Significant Concept Pairs
WordSpace
Association Rule
Domain Specific Taxonomic Relationship
Concept Specification Templates for Non-Taxonomic Relationship
a Domain Ontology (OWL-Lite)

Fig. 4. Quality improvement part



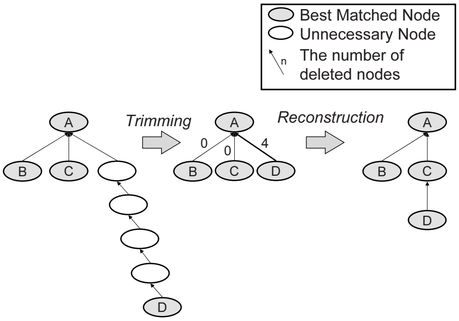


Fig. 5. Matched Result Analysis **Fig. 6.** Trimmed Result Analysis

2.2 Quality Improvement Part

In order to improve the quality of the pre-processed ontology, the quality improvement part works interactively with a user. Fig. 4 shows the procedure of this part. Because the pre-processed taxonomy is constructed from a general ontology, we need to adjust the taxonomy to the specific domain considering an issue called Concept Drift. It means that the position of particular concepts changes depending on the domain. For concept drift management, DODDLE-R applies two strategies: Matched Result Analysis (Fig. 5) and Trimmed Result Analysis (Fig. 6).

In Matched Result Analysis, the system divides the taxonomy into PABs (PAths including only Best matched concepts) and STMs (SubTrees that includes best-matched concepts and other concepts and so can be Moved) and indicates on the screen. PABs are paths that include only best-matched concepts that have senses suitable for the given domain. STMs are subtrees of which root is an internal concept of WordNet and its subordinates are all best-matched concepts. Because the sense of an internal concept has not been identified by a user yet, STMs may be moved to other places for the concept adjustment to the domain. In addition, for Trimmed Result Analysis, the system counts the number of internal concepts when the part was trimmed. By considering this number as the original distance between those two concepts, the system indicates to move the lower concept to other places.

As a facility for related concept pair discovery, there are functions that allow users to attempt some ways to improve the quality of extracted concept pairs through trial and error by changing parameters of statistic methods. Users can re-adjust the parameters of WordSpace and association rule algorithm and check the result. After that, the system generates “Concept Specification Templates” from by using the results. It consists of some concept pairs which have considerable relationship considering the result value of statistic methods.

By referring to the constructed domain specific taxonomic relationship and the “Concept Specification Templates”, a user develops a domain ontology.

3 Implementation

In this section, we describe the system architecture from the aspect of system implementation. DODDLE-R support environment for ontology construction is realized in conjunction with MR^3 (Meta-Model Management based on RDF(S)[11] Revision Reflection) [12]. MR^3 is an RDF(S) graphical editor with meta-model management facility such as consistency checking of classes and a model in which these classes are used as the type of instances. Fig. 7 shows the relationship between DODDLE-R and MR^3 in terms of system implementation. Both MR^3 and DODDLE-R are implemented in Java language (works on Java 2 or higher). MR^3 is implemented using JGraph[13] for RDF(S) graph visualization, and Jena 2 Semantic Web Framework[14] for enabling the use of

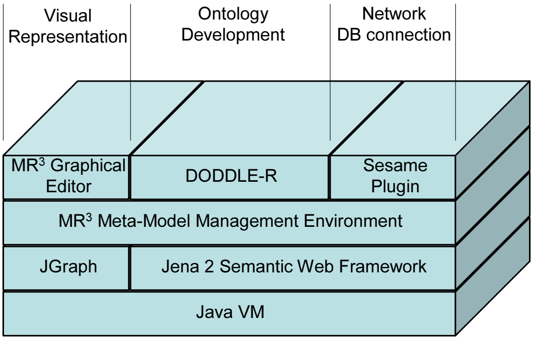


Fig. 7. DODDLE-R architecture

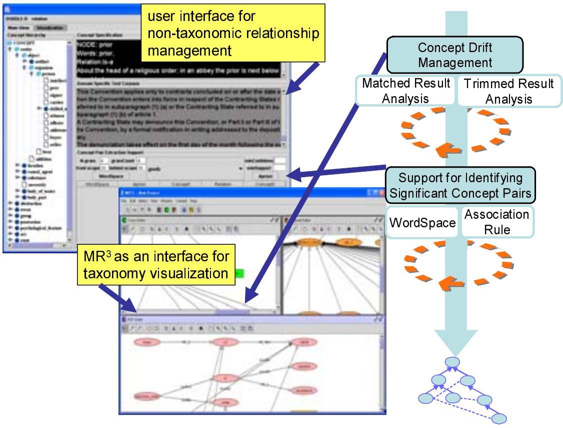
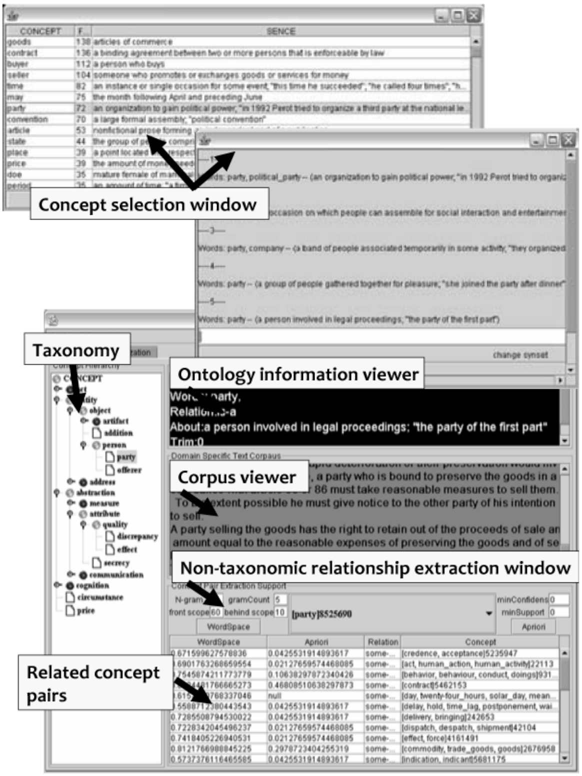

user interface for non-taxonomic relationship management
MR^3 as an interface for taxonomy visualization
Concept Drift Management
Matched Result Analysis
Trimmed Result Analysis
Support for Identifying Significant Concept Pairs
WordSpace
Association Rule

Fig. 8. Quality improvement process with DODDLE-R graphical user interface

Semantic Web standards such as RDF, RDFS, N-triple and OWL. By using these libraries, MR^3 is implemented as an environment for graphical representation of the Semantic Web contents. Additionally, because MR^3 also has plug-in facility to extend its functionality, it can provide some other functions such as the connectivity to Sesame RDF(S) server [15].

On top of MR^3 base environment, DODDLE-R is implemented as a support environment for ontology construction. Fig. 8 depicts the procedure of quality improvement with graphical user interface of the system. DODDLE-R's graphical user interface consists of an ontology information viewer, a corpus viewer and a non-taxonomic relationship acquisition window as in Fig. 9. The ontology information viewer shows the information about particular concepts such as the dictionary definition of the concept, the distance from default root node of ontology. In addition, generated hierarchies are visualized by MR^3 graph editor. On the editor, the system indicates the parts of ontologies which may be modified to make it suitable for the domain according to matched result analysis and trimmed result analysis. The corpus viewer shows the domain specific text corpus which has been referred to acquire related concept pairs by WordSpace and an association rule. When the user clicks a concept on the concept hierarchy, the corpus viewer highlights related terms in the corpus so that the user can see how the term or concept is used in the actual text. The non-taxonomic relationship acquisition window is used for setting parameters for WordSpace and an association rule to apply for the domain specific text corpus in order to generate significantly related concept pairs. For WordSpace, there are parameters such as the gram number (default gram number is four), minimum N-gram count (to extract high-frequency grams only), front scope and behind scope in the text. For an association rule, minimum confidence and minimum support are able to be set by the user.


Concept selection window
Taxonomy
Ontology information viewer
Corpus viewer
Non-taxonomic relationship extraction window
Related concept pairs

CONCEPT	F...	SENCE
goods	138	articles of commerce
contract	136	a binding agreement between two or more persons that is enforceable by law
buyer	112	a person who buys
seller	104	someone who promotes or exchanges goods or services for money
time	82	an instance or single occasion for some event; "this time he succeeded"; "he called four times"; "h...
may	75	the month following April and preceding June
party	72	an organization to gain political power; "in 1992 Perot tried to organize a third party at the national le...
convention	70	a large formal assembly; "political convention"
article	53	nonfictional prose forming
state	44	the group of pe
place	39	a point located
price	39	the amount of mon
doe	35	mature female of man
period	35	an amount of time; "a tim

—1—
Words: party, political_party -- (an organization to gain political power; "in 1992 Perot tried to organiz
occasion on which people can assemble for social interaction and entertainmer
—3—
Words: party, company -- (a band of people associated temporarily in some activity; "they organized
—4—
Words: party -- (a group of people gathered together for pleasure; "she joined the party after dinner"
—5—
Words: party -- (a person involved in legal proceedings; "the party of the first part")

change synset

Concept Hierarchy
- CONCEPT
  - entity
    - object
      - artifact
      - addition
      - person
        - party
        - offerer
    - address
  - abstraction
    - measure
    - attribute
      - quality
        - discrepancy
        - effect
      - secrecy
    - communication
  - cognition
  - circumstance
  - price

Word: party,
Relation: is-a
About:a person involved in legal proceedings; "the party of the first part"
Trim:0

Domain Specific Text Corpaus

, a party who is bound to preserve the goods in a 86 must take reasonable measures to sell them. To extent possible he must give notice to the other party of his intention to sell.
A party selling the goods has the right to retain out of the proceeds of sale an amount equal to the reasonable expenses of preserving the goods and of se

Concept Pair Extraction Support
N-gram gramCount 5 minConfidens 0
front scope 60 behind scope 10 [party]8525690 minSupport 0
WordSpace Apriori

WordSpace	Apriori	Relation	Concept
0.671599627578836	0.0425531914893617	some-...	[credence, acceptance]5235947
0.6901763268659554	0.02127659574468085	some-...	[act, human_action, human_activity]22113
0.7545874211773779	0.10638297872340426	some-...	[behavior, behaviour, conduct, doings]931...
4481766665273	0.46808510638297873	some-...	[contract]5462153
0.615 768337046	null	some-...	[day, twenty-four_hours, solar_day, mean...
0.5588712380443543	0.0425531914893617	some-...	[delay, hold, time_lag, postponement, wai...
0.7285508794530022	0.0425531914893617	some-...	[delivery, bringing]242653
0.7228342045496237	0.02127659574468085	some-...	[dispatch, despatch, shipment]42104
0.7418405226940531	0.02127659574468085	some-...	[effect, force]4161491
0.8121766988845225	0.2978723404255319	some-...	[commodity, trade_goods, goods]2676958
0.5737376116465585	0.0425531914893617	some-...	[indication, indicant]5681175

Fig. 9. A graphical user interface for non-taxonomic relationship management

4 An Example of Ontology Construction with DODDLE-R

In this section, we show a brief example of an ontology construction with DODDLE-R. As a domain specific text corpus for the reference of this ontology construction, we selected the text in CISG (Contracts for the International Sale of Goods)[16] for the particular field of law to compare with the case study which has been done by using DODDLE-II. This corpus is composed of approximately 10,000 words.

4.1 Input Concept Selection

Before starting pre-processing part, a user needs to select some terms as the input. As input of DODDLE-R, the user needs to associate those terms with concepts in WordNet. For example, the user decide which "concept" (or synset) in WordNet is suitable for the term "party" (the noun "party" has 5 senses as in

```
Sense 1
party, political party -- (an organization to gain political power;
 "in 1992 Perot tried to organize a third party at the national level")
       => organization, organisation -- (a group of people who work
          together)
           => social group -- (people sharing some social relation)
               => group, grouping -- (any number of entities (members)
                  considered as a unit)
Sense 2
party -- (an occasion on which people can assemble for social
 interaction and entertainment; "he planned a party to celebrate
 Bastille Day")
       => affair, occasion, social occasion -- (a vaguely specified
          social event; "the party was quite an affair")
           => social event -- (an event characteristic of persons
              forming groups)
               => event -- (something that happens at a given place
                  and time)

...
```

Fig. 10. WordNet concepts for the word "party"

Fig. 10). By referring to the synset and term's definition, the user selects Sence 3 as a concept for the word "party".

4.2 Pre-processing Part

After the user apply selected concepts for the system, a prototype ontology is produced. (A) in Fig. 11 describes the initial model of the taxonomic relationsip. Also related concept pairs are extracted by statistic methods such as WordSpace and assocciation rule by default parameter.

4.3 Quality Improvement Part

After the pre-processing part, there are prototype taxonomy and candidates of concept pairs for concept specification. However, they are just processed automatically and we need to adjust them to actual domain.

(B) in Fig. 11 shows the display of concept drift management. The system indicates some groups of concepts in the taxonomy so that the user can decide which part should be modified.

Also the related concept pairs may be re-extracted by setting the parameters of statistic methods and attempting to get suitable number of concept pairs.

As a result, the user got a domain ontology as in Fig. 12

5 Related Work

Navigli et,al. proposed OntoLearn [17][18], that supports domain ontology construction by using existing ontologies and natural language processing techniques. In their approach, existing concepts from WordNet are enriched and

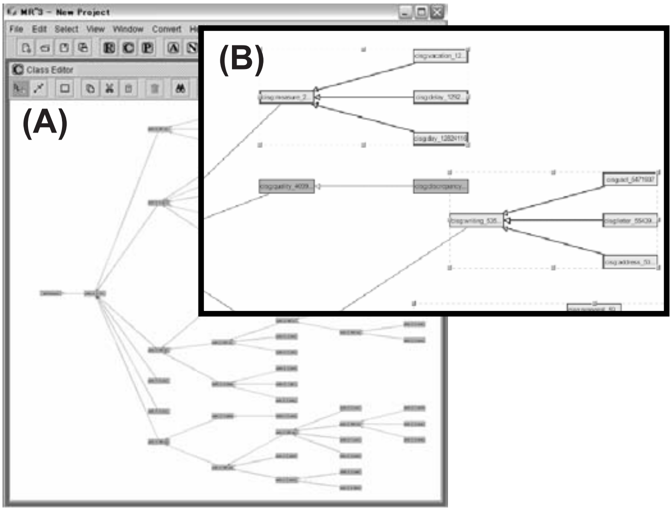


Fig. 11. The initial model of the domain taxonomy (A) and the concept drift management (B)

pruned to fit the domain concepts by using NLP (Natural Language Processing) techniques. They argue that the automatically constructed ontologies are practically usable in the case study of a terminology translation application. However, they did not show any evaluations of the generated ontologies themselves that might be done by domain experts. Although a lot of useful information is in the machine readable dictionaries and documents in the application domain, some essential concepts and knowledge are still in the minds of domain experts. We did not generate the ontologies themselves automatically, but suggests relevant alternatives to the human experts interactively while the experts' construction of domain ontologies. In another case study [19], we had an experience that even if the concepts are in the MRD (Machine Readable Dictionary), they are not sufficient to use. In the case study, some parts of hierarchical relations are counterchanged between the generic ontology (WordNet) and the domain ontology, which are called "Concept Drift". In that case, presenting automatically generated ontology that contains concept drifts may cause confusion of domain experts. We argue that the initiative should be kept not on the machine, but on the hand of the domain experts at the domain ontology construction phase. This is the difference between our approach and Navigli's. Our human-centered approach enabled us to cooperate with human experts tightly.

From the technological viewpoint, there are two different related research areas. In the research using verb-oriented method, the relation of a verb and nouns modified with it is described, and the concept definition is constructed from this information (e.g. [20]). In [21], taxonomic relationships and Subcategorization Frame of verbs (SF) are extracted from technical texts using a machine learning

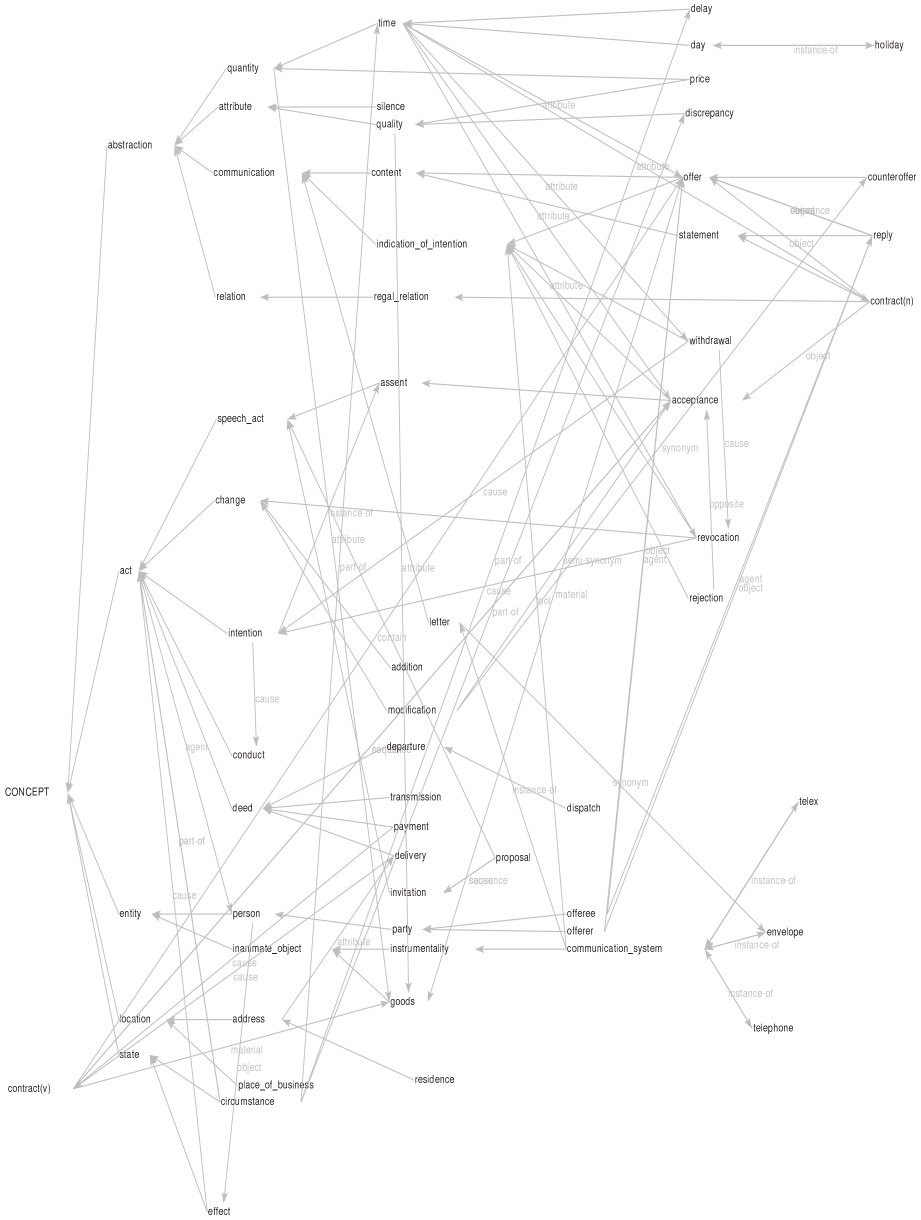


Fig. 12. Constructed CISG ontology

method. The nouns in two or more kinds of different SF with the same frame-name and slot-name are gathered as one concept, base class. And ontology with only taxonomic relationships is built by carrying out clustering of the base class further. Moreover, in parallel, Restriction of Selection (RS) which is slot-value in SF is also replaced with the concept with which it is satisfied instantiated SF. However, proper evaluation is not yet done. Since SF represents the syntactic relationships between verb and noun, the step for the conversion to non-taxonomic relationships is necessary.

On the other hand, in ontology learning using data-mining method, discovering non-taxonomic relationships using an association rule algorithm is proposed by [22]. They extract concept pairs based on the modification information between terms selected with parsing, and made the concept pairs a transaction.

By using heuristics with shallow text processing, the generation of a transaction more reflects the syntax of texts. Moreover, RLA, which is their original learning accuracy of non-taxonomic relationships using the existing taxonomic relations, is proposed. The concept pair extraction method in our paper does not need parsing, and it can also run off context similarity between the terms appeared apart each other in texts or not mediated by the same verb.

6 Conclusion and Future Work

In this paper, we presented a support environment for ontology construction named DODDLE-R, which is aiming at becoming a total support environment for user-centered on-the-fly ontology construction. Its main principle is that high-level support for users through interaction and low dependence on automatic machine processing. First, a user identifies the input concepts by associating WordNet concepts with terms extracted from a text corpus. Then, pre-processing part generates the basis of ontology in the forms of taxonomy and related concept pairs, by referring to WordNet as an MRD and a domain specific text corpus. The quality improvement part provides management facilities for concept drift in the taxonomy and identifying significant concept pairs in extracted related concept pairs. In these management, MR^3 provides significant visualization support for the user in graph representation of ontologies. As a case study, we have constructed an ontology in law domain by exploiting articles in CISG as a domain specific text corpus. Comparing with former ontology construction study with DODDLE-II, even though the first step, input concept selection phase, takes time, other phases are processed fairly well because of the re-organized system architecture and the improved user interface in conjunction with MR^3 . Finally, the user constructed a law domain ontology by interactive support of DODDLE-R and produced an OWL-Lite file, which is able to put on public as a Semantic Web ontology.

We plan further improvement of DODDLE-R to be more flexible ontology development environment. At this point, the user interface of DODDLE-R is not completely supports users' trial and error (in other words, go forward and come back to particular phases of ontology construction seamlessly) in ontology construction. Since we believe that the user interface is one of the most important facilities of support tool for ontology construction, it should be improved to the point of supporting the user seamlessly. In addition, although DODDLE-R extracts domain concepts from text corpus, the extracted terms might be suitable not for concepts (classes) but for relationships (properties) or instances (individuals). For example, the term ”time” may be concept or property (or other kind of attributes). Because the collaboration with MR^3 realized total management of OWL classes, properties and instances (by its editors for each in sub windows and its meta-model management facility), DODDLE-R may be able to support the construction of not only ontologies, but also models, which consist of individuals and their relationships (properties). Furthermore, we plan to implement import facility of other statistic methods. Although DODDLE-R does not emphasize the function in pre-processing part, it would be better to prepare the import facility of other methods. For instance, there is a machine learning software Weka [23], and it contains several machine learning algorithms, which may be suitable for extracting related concept pairs from text corpus. If we look at quality improvement part of DODDLE-R, there may be many additional functions. For instance, for related concept pair extraction by statistic methods, a line graph window is suitable for showing the result of applying statistic methods, also to check the current status of recall and precision. Additionally, in terms of adaptation to the Semantic Web standards, the import and export support of other ontology languages, such as DAML+OIL, must be helpful for interoperability across other ontology tools.

Acknowledgements

This work is supported by Takeshi Morita, a creator of MR^3 and a student in Shizuoka University, Japan.

References

[1] Berners-Lee, T., Hendler, J., Lassila, O.: The Semantic Web. Scientific American (2001)

[2] Gruber, T.: Ontolingua: A Mechanism to Support Portable Ontologies. Version 3.0 TR, KSL (1992)

[3] Heijst, G.V.: The Role of Ontologies in Knowledge Engineering. Dr.thesis, University of Amsterdam (1995)

[4] Ding, Y., Foo, S.: Ontology Research and Development, Part 1 – a Review of Onlotogy. Journal of Information Science (2002) pp.123–136

[5] Lassila, O., Swick, R.R.: Resource Description Framework(RDF) Model and Syntax Specification (1999) http://www.w3.org/RDF/.

[6] Sugiura, N., et al.: A Domain Ontology Engineering Tool with General Ontologies and Text Corpus. Proceedings of the 2nd Workshop on Evaluation of Ontology based Tools (2003) pp.71–82

[7] Michael K. Smith, C.W., McGuinness, D.L.: OWL Web Ontology Language Guide (2004) http://www.w3.org/TR/owl-guide/.

[8] G.A.Miller: WordNet: A Lexical Database for English. ACM (1995) pp.39–41

[9] Marti A. Hearst, H.S.: Customizing a Lexicon to Better Suit a Computational Task. Corpus Processing for Lexical Acquisition (1996) pp.77–96

[10] Agrawal, R., Srikant, R.: Fast algorithms for mining association rules. Proceedings of VLDB Conference (1994) pp.487–499

[11] Brickley, D., Guha, R.: RDF Vocabulary Description Language 1.0: Rdf Schema. W3C Proposed Recommendation (2003) http://www.w3.org/TR/2004/REC-rdf-schema-20040210/.

[12] Noriaki Izumi, Takeshi Morita, N.F., Yamaguchi, T.: RDF-based Meta-Model Management Environment. Proceedings of The 6th SANKEN (ISIR) International Symposium (2003)

[13] Alder, G.: Jgraph. (2003) http://www.jgraph.com.

[14] HP Labs: Jena Semantic Web Framework. (2003)
http://jena.sourceforge.net/downloads.html.

[15] Jeen Broekstra, A.K., Harmelen, F.V.: Sesame: A Generic Architecture for Storing and Querying RDF and RDF Schema. Towards the Semantic Web (2002) pp.71–88 http://sesame.aidministrator.nl.

[16] Sono, K., Yamate, M.: United Nations Convention on Contracts for the International Sale of Goods. Seirin Shoin (1993)

[17] Navigli, R., Paola Velardi: Automatic Adaptation of WordNet to Domains. Proceedings of International Workshop on Ontologies and Lexical Knowledge Bases (2002)

[18] P. Velardi, M.M., Fabriani, P.: Using Text Processing Techniques to Automatically enrich a Domain Ontology. Proceedings of ACM Conf. On Formal ontologies and Information Systems (ACM FOIS) (2001) pp.270–284

[19] Yamaguchi, T.: Constructing domain ontologies based on concept drift analysis. Proceedings of the IJCAI99 Workshop on Ontologies and Problem Solving methods(KRR5) (1999)

[20] Hahn, U., Schnattingerg, K.: Toward text knowledge engineering. AAAI-98 proceedings (1998) pp.524–531

[21] Faure, D., Nédellec, C.: Knowledge Acquisition of Predicate Argument Structures from Technical Texts. Proceedings of International Conference on Knowledge Engineering and Knowledge Management (1999)

[22] Maedche, A., Staab, S.: Discovering Conceptual Relations from Text. Proceedings of 14th European Conference on Artificial Intelligence (2000) pp.321–325

[23] Weka: Machine Learning Software in Java (2004)
http://www.cs.waikato.ac.nz/˜ml/weka/index.html.