# A Domain Ontology Engineering Tool with General Ontologies and Text Corpus

Naoki SUGIURA[1], Masaki KUREMATSU[2], Naoki FUKUTA[1], Noriaki IZUMI[3], and Takahira YAMAGUCHI[1]

[1] Department of Computer Science, Shizuoka University
3-5-1 Johoku, Hamamatsu, Shizuoka, JAPAN
{sugiura, fukuta, yamaguti}@ks.cs.inf.shizuoka.ac.jp
http://panda.cs.inf.shizuoka.ac.jp

[2] Iwate Prefectual University, 152-52 Takizawa-aza-sugo, Takizawa, Iwate, JAPAN
kure@soft.iwate-pu.ac.jp

[3] CARC, National Institute of AIST, 2-41-6 Tokyo Waterfront, Aomi, Koto-ku, Tokyo, JAPAN
niz@ni.aist.go.jp

**Abstract.** In this paper, we describe how to exploit a machine-readable dictionary (MRD) and domain-specific text corpus in supporting the construction of domain ontologies that specify taxonomic and non-taxonomic relationships among given domain concepts. In building taxonomic relationships (hierarchical structure) of domain concepts, some of them can be extracted from an MRD with marked subtrees that may be modified by a domain expert, using matching result analysis and trimmed result analysis. We construct concept specification templates (non-taxonomic relationships of domain concepts) that come from pairs of concepts extracted from text corpus, using WordSpace and an association rule algorithm. Through case studies, we make sure that our system can work to support the process of constructing domain ontologies.

## 1 Introduction

Although ontologies have been very popular in many application areas (e.g. Semantic Web), we still face the problem of high cost associated with building up them manually. In particular, since domain ontologies have the meaning specific to application domains, human experts have to make huge efforts for constructing them entirely by hand. In order to reduce the costs, automatic or semi-automatic methods have been proposed using knowledge engineering techniques and natural language processing ones[1]. However, most of these environments facilitate the construction of only a hierarchically-structured set of domain concepts, in other words, taxonomic conceptual relationships. For example, DODDLE[2] developed by us uses a machine-readable dictionary (MRD) to support a user to construct concept hierarchy only.

In this paper, we extend DODDLE into DODDLE II that constructs both taxonomic and non-taxonomic conceptual relationships, exploiting WordNet[3] and domain-specific text corpus with the automatic analysis of lexical co-occurrence statistics and an association rule algorithm[4]. Furthermore, we evaluate how DODDLE II works in the field of law, CISG (the Contracts for the International Sale of Goods), and in the field of business, xCBL (XML Common Business Library). The empirical results show us that DODDLE II can support a domain expert in constructing domain ontologies.

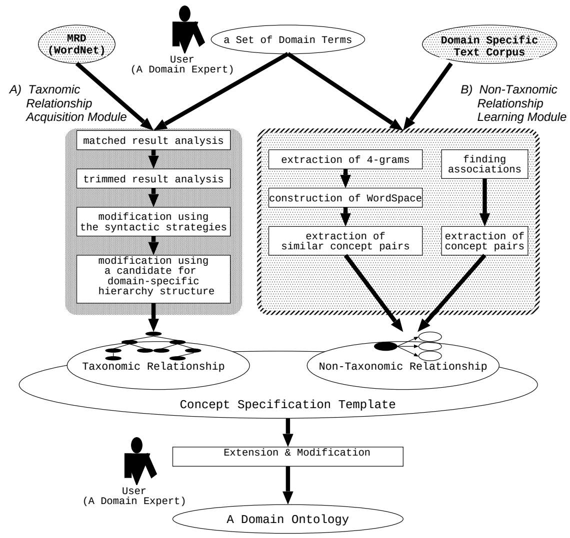


**Fig. 1.** DODDLE II overview

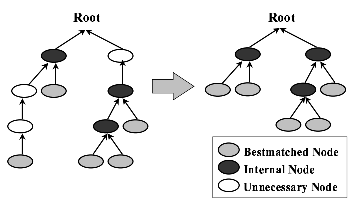


**Fig. 2.** Trimming Process

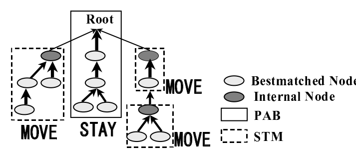


**Fig. 3.** Matched Result Analysis

## 2 DODDLE II: A Domain Ontology Rapid Development Environment

### 2.1 Overview

Figure 1 shows the overview of DODDLE II, "*a Domain Ontology rapiD DeveLopment Environment*". DODDLE II tries to construct a domain ontology from a set of domain terms given by a human expert using the following two components: Taxonomic Relationship Acquisition module (TRA module) using WordNet and Non-Taxonomic Relationship Learning module (NTRL module) using text corpus. WordNet is a existing MRD used in some systems.
A) TRA module tries to support a user in constructing taxonomic relationship (concept hierarchy) using Word Net.
B) NTRL module extracts the pairs of terms that should be related by some relationships from text corpus, analyzing lexical co-occurrence statistics, based on WordSpace[5] and an associate rule algorithm.

We can build concept specification templates by putting together taxonomic and non-taxonomic relationships for the input domain terms. The relationships should be identified in the interaction with a human expert.

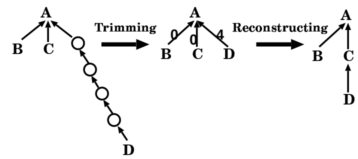


**Fig. 4.** Trimmed Result Analysis

### 2.2 Taxonomic Relationship Acquisition

First of all, TRA module does “spell match” between input domain terms and WordNet. The “spell match” links these terms to WordNet. Thus the initial model from the “spell match” results is a hierarchically structured set of all the nodes on the path from these terms to the root of WordNet. However, the initial model has unnecessary internal terms (nodes) and they do not contribute to keep topological relationships among matched nodes, such as parent-child relationship and sibling relationship. So we get a trimmed model by trimming the unnecessary internal nodes from the initial model (see Fig.2). After getting the trimmed model, TRA module refines it by interaction with a domain expert, using Matched result analysis (see Fig.3) and Trimmed result analysis (see Fig.4). TRA module divides the trimmed model into a PAB (a PAth including only Best spell-matched nodes) and an STM (a Subtree that includes best spell-matched nodes and other nodes and so can be Moved) based on the distribution of best-matched nodes. A PAB is a path that includes only best-matched nodes that have the senses good for given domain specificity. Because all nodes have already been adjusted to the domain in PABs, PABs can stay in the trimmed model. An STM is such a subtree that an internal node is a root and the subordinates are only best-matched nodes. Because internal nodes have not been confirmed to have the senses good for a given domain, an STM can be moved in the trimmed model.

In order to refine the trimmed model, DODDLE II can use trimmed result analysis. Taking some sibling nodes with the same parent node, there may be big differences about the number of trimmed nodes between them and the parent node. When such a big difference comes up on a subtree in the trimmed model, it is better to change the structure of it. DODDLE II asks a human expert whether the subtree should be reconstructed. Based on the empirical analysis, the subtrees with two or more differences may be reconstructed.

Finally, DODDLE II completes taxonomic relationships of the input domain terms manually from the user.

### 2.3 Non-Taxonomic Relationship Learning

NTRL module almost comes from WordSpace, which derives lexical co-occurrence information from a large text corpus and is a multi-dimension vector space (a set of vectors). The inner product between two word vectors works as the measure of their semantic relatedness. When two words’ inner product is beyond some upper bound, there are possibilities to have some non-taxonomic relationship between them. NTRL module also uses an association rule algorithm to find associations between terms in text corpus. When an association rule between terms exceeds user-defined thresholds, there are possibilities to have some non-taxonomic relationships between them.

**Construction of WordSpace** WordSpace is constructed as shown in Fig.5.

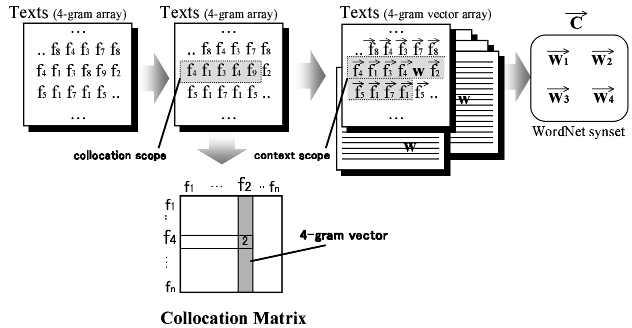


**Fig. 5.** Construction Flow of WordSpace

*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-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 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 II 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.

**Constructing and Modifying Concept Specification Templates** A set of similar concept pairs from WordSpace and term pairs from the association rule algorithm becomes concept specification templates. Both of the concept pairs, whose meaning is similar (with taxonomic relation), and has something relevant to each other (with non-taxonomic relation), are extracted as concept pairs with above-mentioned methods. However, by using taxonomic information from TRA module with co-occurrence information, DODDLE II distinguishes the concept pairs which are hierarchically close to each other from the other pairs as *TAXONOMY*. A user constructs a domain ontology by considering the relation with each concept pair in the concept specification templates, and deleting unnecessary concept pairs.

## 3 Case Studies

In order to evaluate how DODDLE II is doing in a practical field, case studies have been done in particular field of law and business. The particular field of law is called "Contracts for the International Sale of Goods"(CISG)[6] and the particular field of business is called "XML Common Business Library"(xCBL)[7]. DODDLE II is being implemented on Perl/Tk now. Figure 6 shows s snapshot.

### 3.1 A Case Study in the Law Field

**Input terms in the Case Study with CISG**

A layer is a user in this case study. Table 1 shows significant 46 legal terms extracted by the user from CISG Part-II. He gave them DODDLE II as input terms.

**Taxonomic Relationship Aqcuisition**

Table 2 shows the number of concepts in each model under taxonomic relationship acquisition and Fig.7 shows the concept hierarchy constructed by the user using DODDLE II. Table 3 shows the evaluation of two strategies by the user. Precision is the percentage of extracted subtrees that were accepted, recall per path is the percentage of extracted paths that were accepted and recall per subtree is the percentage of extracted subtrees that were accepted. Precision and both recalls are less than 0.3 and are not good.

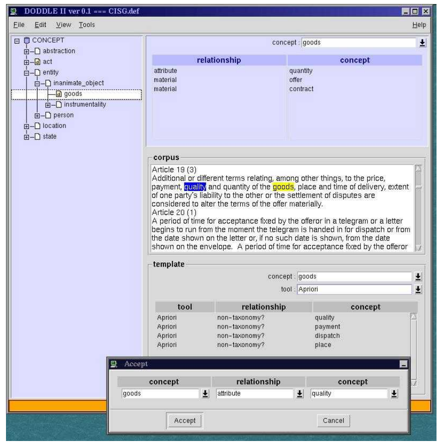


**Fig. 6.** The Ontology Editor

**Table 1.** Significant 46 Concepts in CISG Part-II

| | | | | | |
|---|---|---|---|---|---|
| acceptance | act | addition | address | assent | circumstance |
| communication | conduct | contract | counteroffer | day | delay |
| delivery | discrepancy | dispatch | effect | envelope | goods |
| holiday | indication | intention | invitation | letter | modification |
| offer | offeree | offerer | party | payment | person |
| placeofbusiness | price | proposal | quality | quantity | rejection |
| reply | residence | revocation | silence | speechact | system |
| telephone | telex | time | transmission | withdraw | |

But about 70 % of the concept hierarchy (taxonomic relationships) were constructed with TRA module support and about half portion of them results in the information extracted from WordNet. Therefore we evaluated TRA model worked well in this case study. The detail of this case study is described in [2].

### Non-Taxonomic Relationship Learning

*Construction of WordSpace* High-frequency 4-grams were extracted from CISG (about 10,000 words) standard form conversion removed duplication, and 543 kinds of 4-grams were obtained. In order to keep density of a collocation matrix high, the extraction frequency of 4-grams must be adjusted according to the scale of text corpus. As CISG is the comparatively small-scale text, the extraction frequency was set as 7 times this case. In order to construct a context vector, a sum of 4-gram vectors around appearance place circumference of each of 46 concepts was calculated. In order to construct a context scope from some 4-grams, it consists of putting together 60 4-grams before the 4-gram and 10 4-grams after the 4-grams independently of length of a sentence. For each of 46 concepts, the sum of context vectors in all the appearance places of the concept in CISG was calculated, and the vector representations of the concepts were obtained. The set of these vectors is used as WordSpace to extract concept pairs with context similarity. Having calculated the similarity from the inner product for the 1,035 concept pairs which are all the combination of 46 concepts, and having used threshold as 0.87, 77 concept pairs were extracted.

**Table 2.** the Change of the Number of Concepts under Taxonomic Relationship Acquisiion

| Model | Input Terms | Initial Model | Triimed Model | Concept Hierarchy |
|---|---|---|---|---|
| # Concept | 46 | 133 | 56 | 61 |

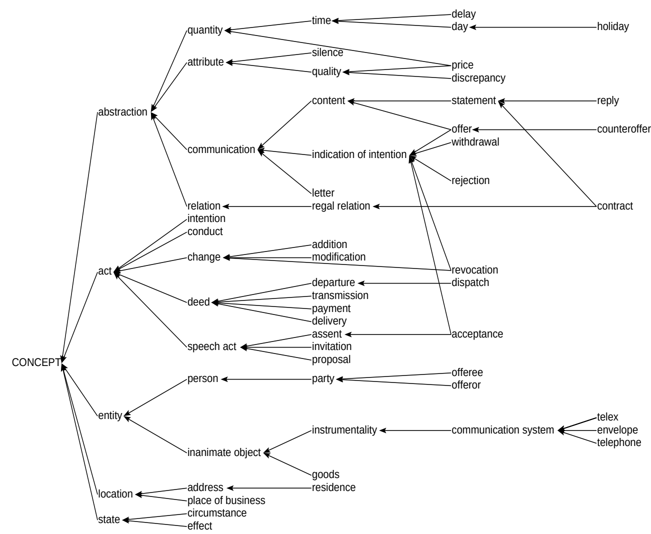


**Fig. 7.** Domain Concept Hierarchy of CISG Part II

**Table 3.** Precision and Recall in the Case Study with CISG

| | Precision | Recall per Path | Recall per Subtree |
|---|---|---|---|
| Matched Result | 0.25(4/16) | 0.23(5/21) | 0.19(4/19) |
| Trimmed Result | 0.3(3/10) | 0.3(6/20) | 0.15(3/20) |

*Finding Associations between Input Terms* In this case, DODDLE II extracted 55 pairs of terms from text corpus using the above-mentioned association rule algorithm. There are 15 pairs out of them in a set of similar concept pairs extracted using WordSpace.

*Constructing and Modifying Concept Specification Templates* Concept specification templates were constructed from two sets of concept pairs extracted by WordSpace and Associated Rule algorithm. In concept specification templates, such a concept is distinguished as *TAXONOMY* relation. As taxonomic and non-taxonomic relationships may be mixed in the list based on only context similarity, the concept pairs which may be concerned with non-taxonomic relationships are obtained by removing the concept pairs with taxonomic relationships. After the user thought concept definitions, the user modified concept specification templates. Figure 8.(A) shows concept specification templates about the concept "goods". Figure 8.(B) shows the definition of the concept "goods" constructed from consideration of concept pairs in the templates.

*Evaluation of Results of NTRL module* The user evaluated the following two sets of concept pairs: one is extracted by WordSpace(WS) and the other is extracted by Association Rule algorithm (AR). Figure 9 shows three different sets of concept pairs from the user,

| (A) Concept Specific Template | |
|---|---|
| *non-TAXONOMY?* | : **quality** |
| *non-TAXONOMY?* | : **quantity** |
| *non-TAXONOMY?* | : **contract** |
| *non-TAXONOMY?* | : **act** |
| *non-TAXONOMY?* | : **delivery** |
| *non-TAXONOMY?* | : **effect** |
| *non-TAXONOMY?* | : **party** |
| *non-TAXONOMY?* | : **payment** |
| *non-TAXONOMY?* | : **person** |
| *non-TAXONOMY?* | : **price** |
| *non-TAXONOMY?* | : **time** |

⇒

| (B) Concept Definition | |
|---|---|
| ATTRIBUTE | : **quality** |
| ATTRIBUTE | : **quantity** |
| MATERIAL | : **contract** |
| MATERIAL | : **offer** |

**Fig. 8.** The Concept Specification Templates and Concept Definition for "goods"

**Table 4.** Evaluation by the User with Legal Knowledge

| | WordSPace (WS) | Association Rules (AR) | The Join of WS and AR |
|---|---|---|---|
| # Extracted concept pairs | 77 | 55 | 117 |
| # Accepted concept pairs | 18 | 13 | 27 |
| # Rejected concept pairs | 59 | 42 | 90 |
| Precision | 0.23(18/77) | 0.24(13/55) | 0.23(27/117) |
| Recall | 0.38(18/48) | 0.27(13/48) | 0.56(27/48) |

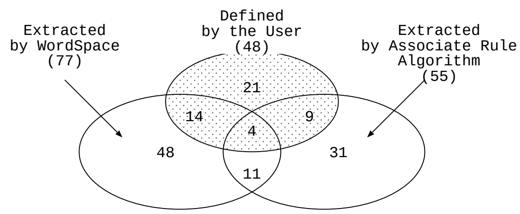


**Fig. 9.** Three Different Sets of Concept Pairs from User, WS and AR

WS and AR. Table 4 shows the details of evaluation by the user, computing precision and recall. Precision is the percentage of concept pairs accepted by a user that were extracted by DODDLE II. Recall is the percentage of concept pairs extracted by DODDLE II that were defined by a user. Looking at the field of Precision in Table 4, there is almost no differences among three kinds of results from WS,AR and the join of them. However, looking at the field of Recall in Table 4, the recall from the join of WS and AR is higher than that from each WS and AR, and then goes over 0.5.

### 3.2 A Case Study in the Business Field

**Input terms in the Case Study with xCBL**

Table 5 shows input terms in this case study. They are 57 business terms extracted by a user from xCBL Document Reference. The user is not a expert but has business knowledge.

**Taxonomic Relationship Acquisition**

Table 6 shows the number of concept pairs in each model under taxonomic relationship acquisition and table 7 shows the evaluation of two strategies by the user. The recall per subtree is more than 0.5 and is good. The precision and the recall per path are less than 0.3 and are not so good, but about 80 % portion of taxonomic relationships were

**Table 5.** Significant 57 Concepts in xCBL

| | | | | |
|---|---|---|---|---|
| acceptance | agreement | auction | availability | business |
| buyer | change | contract | customer | data |
| date | delivery | document | exchange rate | financial institution |
| foreign exchange | goods | information | invoice | item |
| line item | location | marketplace | message | money |
| order | organization | partner | party | payee |
| payer | payment | period of time | price | process |
| product | purchase | purcahse agreement | purchase order | quantity |
| quotation | quote | receipt | rejection | request |
| resource | response | schedule | seller | service |
| shipper | status | supplier | system | third party |
| transaction | user | | | |

**Table 6.** The Change of the Number of Concepts under Taxonomic Relationship Acquisision

| Model | Input Terms | Initial Model | Triimed Model | Concept Hierarchy |
|---|---|---|---|---|
| # Concept | 57 | 152 | 83 | 82 |

**Table 7.** Precision and Recall in the Case Study with xCBL

| | Precision | Recall per Path | Recall per Subtree |
|---|---|---|---|
| Matched Result | 0.2(5/25) | 0.29(5/17) | 0.71(5/7) |
| Trimmed Result | 0.22(2/9) | 0.13(2/15) | 0.5(2/4) |

constructed with TRA module support. We evaluated TRA module worked well in this case study.

**Non-Taxonomic Relationship Learning**

*Construction of WordSpace* High-frequency 4-grams were extracted from xCBL Document Description (about 2,500 words) standard form conversion removed duplication, and 1240 kinds of 4-grams were obtained. In order to keep density of a collocation matrix high, the extraction frequency of 4-grams must be adjusted according to the scale of text corpus. As xCBL text are shorter than CISG text, the extraction frequency was set as 2 times this case. In order to construct a context vector, a sum of 4-gram vectors around appearance place circumference of each of 57 concepts was calculated. In order to construct a context scope from some 4-grams, it consists of putting together 10 4-grams before the 4-gram and 10 4-grams after the 4-grams independently of length of a sentence. For each of 57 concepts, the sum of context vectors in all the appearance places of the concept in xCBL was calculated, and the vector representations of the concepts were obtained. The set of these vectors is used as WordSpace to extract concept pairs with context similarity. Having calculated the similarity from the inner product for concept pairs which is all the combination of 57 concepts, 40 concept pairs were extracted.

*Finding Associations between Input Terms* DODDLE II extracted 39 pairs of terms from text corpus using the above-mentioned association rule algorithm. There are 13 pairs out of them in a set of similar concept pairs extracted using WordSpace. Then, DODDLE II constructed concept specification templates from two sets of concept pairs extracted by WordSpace and Associated Rule algorithm. However, the user didn't have enough time to modify them and didn't finish to modify them.

*Evaluation of Results of NTRL module* The user evaluated the following two sets of concept pairs: one is extracted by WS(WordSpace) and the other is extracted by AR(Association Rule algorithm). Figure 10 shows two different sets of concept pairs from WS and AR. It also shows portion of extracted concept pairs that were accepted by the user. Table 8 shows the details of evaluation by the user, computing precision only. Because the user didn't define concept definition in advance, we can not compute recall. Looking at the field of precision in Table 8, the precision from WS is higher than others. Most of concept pairs which have relationships were extracted by WS. The percentage is about 77%(30/39). But there are some concept pairs which were not extracted by WS. Therefore taking the join of WS and AR is the best method to support a user to construct non-taxonomic relationships.

### 3.3 Results and Evaluation of Case Studies

In regards to support in constructing taxonomic relationships, the precision and recall are less than 0.3 in both case studies and there is almost no difference. Generally, 70 %

**Table 8.** Evaluation by the User with xCBL definition

| | WordSPace (WS) | Association Rules (AR) | The Join of WS and AR |
|---|---|---|---|
| # Extracted concept pairs | 40 | 39 | 66 |
| # Accepted concept pairs | 30 | 20 | 39 |
| # Rejected concept pairs | 10 | 19 | 27 |
| Precision | 0.75(30/40) | 0.51(20/30) | 0.59(39/66) |

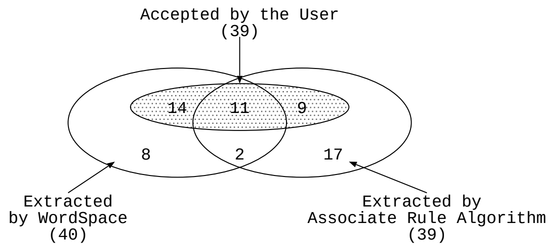


**Fig. 10.** Two Difference Sets of Concept Pairs from WS and AR and Concept Sets have Relationships

or more support comes from TRA module. About more than half portion of the final domain ontology results in the information extracted from WordNet. Because the two strategies just imply the part where concept drift may come up, the part generated by them has low component rates and about 30 % hit rates. So one out of three indications based on the two strategies work well in order to manage concept drift. The two strategies use matched and trimmed results, therefore based on structural information of an MRD only, the hit rates are not so bad. In order to manage concept drift smartly, we may need to use more semantic information that is not easy to come up in advance in the strategies, and we also may need to use domain specific text corpus and other information resource to improve supporting a user in constructing taxonomic relationships.

In regards to construction of non-taxonomic relationships, the precision in the case study with xCBL is good, but the precision in the case study with CISG is less than 0.3 and not good. Generating non-taxonomic relationships of concepts is harder than modifying and deleting them. Therefore, DODDLE II supports the user in constructing non-taxonomic relationships.

After analyzing results of case studies, we have the following problems.

**1. Determination of a Threshold:** Threshold of the context similarity changes in effective value with each domain. It is hard to set up the most effective value in advance.

**2. Specification of a Concept Relation:** Concept specification templates have only concept pairs based on the context similarity, it requires still high cost to specify relationships between them. It is needed to support specification of concept relationships on this system in the future work.

**3. Ambiguity of Multiple Terminology:** For example, the term “transmission” is used in two meanings, “transmission (of goods)” and “transmission (of communication)”, in the article, but DODDLE II considers these terms as the same and creates WordSpace as it is. Therefore constructed vector expression may not be exact. In order to extract more useful concept pairs, semantic specialization of a multisense word is necessary, and it should be considered that the 4-grams with same appearance and different meaning are different 4-grams.

## 4 Related Work

Navigli et,al. proposed OntoLearn [8][9][10], that supports domain ontology construction by using existing ontologies and natural language processing techniques. In their approach, existing concepts from WordNet are enriched and pruned to fit the domain concepts by using NLP 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 [11], we had an experience that even if the concepts are in the MRD, 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. [13]). In [14], taxonomic relationships and Subcategorization Frame of verbs (SF) are extracted from technical texts using a machine learning 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 [12]. 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.

## 5 Conclusions

In this paper, we discussed how to construct a domain ontology using an existing MRD and text corpus. In order to acquire taxonomic relationship, two strategies have been proposed: matched result analysis and trimmed result analysis. Furthermore, to learn non-taxonomic relationships, concept pairs may be related to concept definition, extracted on the basis of the co-occurrence information in text corpus, and a domain ontology is developed by the modification and specification of concept relations with concept specification templates. It serves as the guideline for narrowing down huge space of concept pairs to construct a domain ontology.

It is almost craft-work to construct a domain ontology, and still difficult to obtain the high support rate on system. DODDLE II mainly supports for construction of a concept hierarchy with taxonomic relationships and extraction of concept pairs with non-taxonomic relationships now. However a support for specification concept relationship is indispensable. The future works are as follows: improvement in the scalability of the definition support by learning of heuristics and introduction of the useful datamining method instead of WordSpace, and system integration of taxonomic relationship acquisition module and non-taxonomic relationship learning module (now implementing).

## 6 Acknowledgment

This work has been supported by Naomi Nakaya and Takamasa Iwade (former students of Shizuoka University, Japan).

## References

1. Y. Ding and S.Foo: “Ontology Research and Development, Part 1 – A Review of Onlotogy”, Journal of Information Science, Vol.28, No2, 123 – 136, 2002
2. Rieko Sekiuchi, Chizuru Aoki, Masaki Kurematsu, and Takahira Yamaguchi: “DODDLE : A Domain Ontology Rapid Development Environment”, PRICAI98, 1998
3. C.Fellbaum ed: “Wordnet”, The MIT Press, 1998. see also URL: http://www.cogsci.princeton.edu/~wn/
4. Rakesh Agrawal and Ramakrishnan Srikant :“Fast algorithms for mining association rules,”, Proc. of VLDB Conference, 487–499, 1994
5. Marti A. Hearst and Hinrich Schutze: “Customizing a Lexicon to Better Suit a Computational Task”, in Corpus Processing for Lexical Acquisition edited by Branimir Boguraev & James Pustejovsky, 77–96
6. Kazuaki Sono and Masasi Yamate: United Nations convention on Contracts for the International Sale of Goods, Seirin-Shoin, 1993
7. xCBL.org: http://www.xcbl.org/xcbl40/documentation/ listofdocuments.html
8. R. Navigli and P. Velardi: “Ontology Learning and Its Application to Automated Terminology Translation”, IEEE Intelligent Systems, JANUARY/FEBRUARY, pp.22–31, 2003.
9. Roberto Navigli and Paola Velardi: “Automatic Adaptation of WordNet to Domains”, Proc. of International Workshop on Ontologies and Lexical Knowledge Bases(OntoLex2002), 2002.
10. P. Velardi, M. Missikoff, and P. Fabriani : “Using Text Processing Techniques to Automatically enrich a Domain Ontology”, Proc. of ACM Conf. On Formal ontologies and Information Systems(ACM FOIS), pp.270–284, October 2001.
11. T. Yamaguchi: “Constructing Domain Ontologies Based on Concept Drift Analysis”, Proc. of the IJCAI99 Workshop on Ontologies and Problem Solving methods(KRR5), August 1999.
12. Alexander Maedche and Steffen Staab: “Discovering Conceptual Relations from Text”, ECAI2000, 321–325, 2000
13. Udo Hahn and Klemens Schnattingerg: “Toward Text Knowledge Engineering”, AAAI98, IAAAI-98 proceedings, 524-531, 1998
14. David Faure and Claire Nédellec: “Knowledge Acquisition of Predicate Argument Structures from Technical Texts Using Machine Learning: The System ASIUM”, EKAW’99