Method of Extracting *Is-A* and *Part-Of* Relations Using Pattern Pairs in Mass Corpus ∗

Se-Jong Kim, Yong-Hun Lee, and Jong-Hyeok Lee

Department of Computer Science and Engineering,
Pohang University of Science and Technology (POSTECH),
San 31, Hyoja-dong, Nam-gu, Pohang, 790-784, Republic of Korea
{sejong, yhlee95, jhlee}@postech.ac.kr

Abstract. This paper proposes a method that extracts term pairs satisfying *is-a* relations or *part-of* relations from a mass corpus using pairs of patterns sharing a term. We extracted reliable single patterns and pattern pairs using some term pairs that satisfy the target relation, and extracted reliable term pairs using these patterns. The extracted term pairs were used to extract new single patterns and pattern pairs, and we repeated these steps several times. The proposed method achieved 71.5% accuracy in detecting *is-a* relations and 88% accuracy in detecting *part-of* relations, and extracted 144 new *is-a* relations and 85 new *part-of* relations which could not be extracted using single patterns. These results are useful in constructing an ontology and a thesaurus because these language knowledge bases consist mainly of *is-a* relations and *part-of* relations.

Keywords: Bootstrapping algorithm, shared term, reliability equation, subjective evaluation, morphological analysis.

1 Introduction

An ontology is a language knowledge base that organizes categories of concepts, and defines relations among the categories. The ontology is useful in semantic analysis of natural language processing (NLP) and information retrieval (IR). Methods that construct the ontology fall into two categories: those that combine existing ontologies and thesauruses (Maedche and Staab, 2002) and those that semi-automatically construct the ontology using a mass corpus (Kavalec and Svatek, 2005). The first method focuses on expanding the ontology using existing resources, and the second method focuses on constructing new ontologies. Particularly, constructing the ontology using a mass corpus begins with automatic extraction of semantic relations.

Extraction of semantic relations is a process that extracts term pairs that satisfy a specified target relation. The most common approach in this process is a pattern-based approach for *is-a* and *part-of* relations (Berland and Charniak, 1999; Girju *et al.*, 2003; Hearst, 1992; Pantel and Pennacchiotti, 2006; Ravichandran and Hovy, 2002). This approach regards words appearing between terms as patterns, and extracts term pairs using the patterns. Hearst (1992) extracts *is-a* relations using manual patterns, and proposes a bootstrapping algorithm using seed term pairs. To extract *part-of* relations, Berland and Charniak (1999) measure the reliabilities of terms using the frequencies of terms that have the concept of "whole" or "part," and Girju *et al.* (2003) uses a lexical database (WordNet) and a decision tree. Ravichandran and Hovy (2002) extract semantic relations for various terms in a question answering (QA) system. Pantel and

∗ The work reported in this paper was supported in part by the Korea Science and Engineering Foundation (KOSEF) grant funded by the Korean government (MEST No. 2009-0075211), and in part by the BK 21 Project in 2009.

Pennacchiotti (2006) propose *Espresso* algorithm that uses the reliabilities of terms and patterns, and that has good performance. This algorithm is applied to a coreference resolution (Yang and Su, 2007). However, these methods consider only single sentences to extract patterns, and can not extract pairs of terms in different sentences. In single sentences, the term pairs satisfying *is-a* relations are, in fact, few, and the kind of patterns reflecting each target relation are very restricted. If previous methods use an open-domain corpus, these methods may not be useful.

We propose a method that extracts term pairs satisfying *is-a* relations or *part-of* relations from a mass corpus using pairs of patterns sharing a term, and compare the result with a previous method (Pantel and Pennacchiotti, 2006) that uses single patterns. The proposed method can extract pairs of terms in different sentences in the corpus because this method uses pairs of patterns that can appear in different sentences. This characteristic of pattern pairs will enable extraction of reliable term pairs, and produce good performance.

For the previous method and the proposed method, we evaluate the accuracy of the extracted term pairs, and count the new term pairs which can not be extracted using single patterns. A description of the proposed method is given in Section 2, results are given in Section 3, a discussion is given in Section 4, and a conclusion is given in Section 5.

2 Materials and Methods

2.1 Overview

A “term” is a meaningful noun or noun phrase that does not include articles, and a “pattern” is the set of positions of the terms and the words between the terms. The positions use a variable *X* and *Y*. If a sentence is “A human is an animal,” terms are “human” and “animal,” and a pattern is “*X* is an *Y*.”

Espresso algorithm (Pantel and Pennacchiotti, 2006) uses a bootstrapping algorithm (Hearst, 1992) and has good performance. This algorithm extracts patterns appearing between some prepared term pairs (seed term pairs) that satisfy a specified target relation. The reliabilities of these patterns are measured by the pattern reliability equation (r_P) using *pointwise mutual information* (*pmi*) and *discounting factor* (*df*) (Pantel and Ravichandran, 2004) as:

$$r_P(p) = \frac{\sum_{(x,y)\in XY}\left(pd((x,y),p)\times r_{XY}(x,y)\right)}{|XY|}, \tag{1}$$

where
$$pd((x,y),p) = \frac{pmi((x,y),p)\times df((x,y),p)}{\max_{pmi\times df}}.$$

In (1), p is a pattern; (x, y) is a term pair; XY is a set of term pairs used to extract patterns and r_{XY} is the term pair reliability equation,

where
$$r_{XY}(x,y) = \frac{\sum_{p\in P}\left(pd((x,y),p)\times r_P(p)\right)}{|P|}, \tag{2}$$

where P is a set of cumulative patterns used to extract term pairs.

The patterns are sorted by r_P, and the most reliable pattern is used to extract new term pairs. The reliabilities of new term pairs are also measured by r_{XY}. The reliable term pairs are selected by the reliabilities of term pairs, and these term pairs are used to extract new patterns. *Espresso* repeats these steps several times and accumulates term pairs continually.

2.2 Materials

We constructed experimental text data consisting of sentences with POS tags. The sentences were extracted from Wikipedia[1], because this web site was not a restricted domain and we could easily extract many sentences. We removed some text errors using a text editor (UltraEdit-32) and simple rules. We compiled a mass corpus by gathering 947,625 sentences, and added POS tags to the mass corpus using a morphological analyzer (Stanford tagger[2]).

We coded five programs to process the experimental data. The programs were the term pair extractor, the single pattern extractor, the pattern pair extractor, the set {term pair, single pattern} extractor, and the semantic relation extractor. These were developed using a Java-based development tool (NetBeans IDE 6.1, Java 1.5).

2.3 Methods

We extracted *is-a* relations and *part-of* relations using pairs of patterns sharing a term. Our methods were based on the *Espresso* algorithm. To do this work, we prepared seed term pairs, extracted necessary data from the corpus, and modified reliability equations to apply pattern pairs to the *Espresso* algorithm.

2.3.1. Preparing seed term pairs

We prepared 10 seed term pairs for each relation to extract single patterns and pattern pairs in the first iteration (Table 1). Each term was a singular noun and had a POS tag.

Table 1: Seed term pairs with POS tag for two semantic relations.

Semantic relation	
***Is-a* relation**	***Part-of* relation**
{wheat/nn, crop/nn}	{memory/nn, computer/nn}
{miami/nnp, city/nn}	{drawer/nn, desk/nn}
{shark/nn, fish/nn}	{roof/nn, house/nn}
{apple/nn, fruit/nn}	{hydrogen/nn, water/nn}
{man/nn, human/nn}	{head/nn, body/nn}
{milk/nn, beverage/nn}	{branch/nn, tree/nn}
{flower/nn, plant/nn}	{wing/nn, airplane/nn}
{computer/nn, machine/nn}	{sea/nn, earth/nn}
{desk/nn, table/nn}	{player/nn, team/nn}
{noise/nn, sound/nn}	{wheel/nn, car/nn}

2.3.2. Extracting data

We extracted term pairs, single patterns, pattern pairs, sets of {term pair, single pattern} from the mass corpus, and measured the frequency of each. The term pairs were extracted when a pair of terms satisfying the regular expression (3) appeared in one sentence.

$$((adj.\,|\,noun)*(noun)(prep.))?(adj.\,|\,noun)*(noun) \tag{3}$$

In (3), the * operator indicates there are zero or more preceding element, and the ? operator indicates there is zero or one preceding element. The single patterns were extracted when words appeared between the extracted terms. When two terms appearing before or after two single patterns were the same term (Term *C*), the two single patterns were regarded as one pattern pair

[1] Wikipedia, http://www.wikipedia.org. This web site provides backup dumps of wikitext source. We used the backup dump in 2008-06-13 as experimental text data.

[2] Stanford tagger, http://nlp.stanford.edu/software/tagger.shtml.

(Figure 1). Pattern pairs were grouped into four types by positions of the terms: {*X* ⋯ *C*, *C* ⋯ *Y*}; {*X* ⋯ *C*, *Y* ⋯ *C*}; {*C* ⋯ *X*, *C* ⋯ *Y*}; {*C* ⋯ *X*, *Y* ⋯ *C*}.

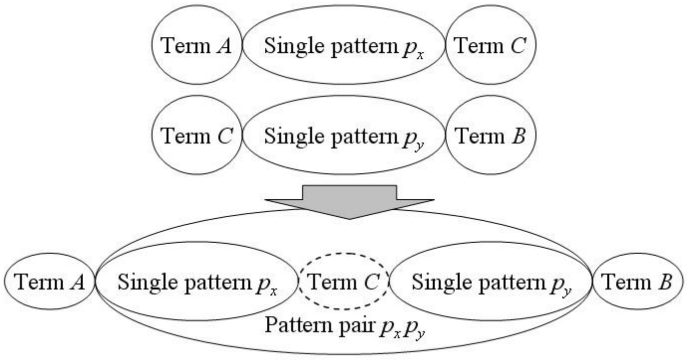

Figure 1: Changing two single patterns into one pattern pair. Term *C* appears at the end of pattern p_x, and the beginning of pattern p_y, so p_x and p_y are merged into a pattern pair.

2.3.3. Modifying reliability equations

We proposed two methods that modified the initial reliability equations (1) and (2). Method Ⅰ modified *pmi* and *df*, and Method Ⅱ used combining statistical values for each pattern of the pair and the weight of the shared term to modify (1) and (2).

2.3.3.1. Method Ⅰ

To propose reliability equations that could process pattern pairs, we modified *pmi* and *df* as:

$$\begin{aligned} pmi\left((x,y), p_{pair}\right) &= \log \frac{|x, p_{pair}, y| \times |*,*,*|}{|x,*,y| \times |*, p_{pair}, *|} \\ &= \log \frac{\sum_{c \in T}\left(|x, p_x, c| \times |c, p_y, y|\right) \times \sum_{c' \in T}\left(|*,*,c'| \times |c',*,*|\right)}{\sum_{c'' \in T}\left(|x,*,c''| \times |c'',*,y|\right) \times \sum_{c''' \in T}\left(|*, p_x, c'''| \times |c''', p_y, *|\right)}, \end{aligned} \tag{4}$$

$$\begin{aligned} & df\left((x,y), p_{pair}\right) \\ &= \frac{|x, p_{pair}, y|}{|x, p_{pair}, y| + 1} \times \frac{\min\left(|x,*,y|, |*, p_{pair}, *|\right)}{\min\left(|x,*,y|, |*, p_{pair}, *|\right) + 1} \\ &= \frac{\sum_{c \in T}\left(|x, p_x, c| \times |c, p_y, y|\right)}{\sum_{c \in T}\left(|x, p_x, c| \times |c, p_y, y|\right) + 1} \times \frac{\min\left(\sum_{c' \in T}\left(|x,*,c'| \times |c',*,y|\right), \sum_{c'' \in T}\left(|*, p_x, c''| \times |c'', p_y, *|\right)\right)}{\min\left(\sum_{c' \in T}\left(|x,*,c'| \times |c',*,y|\right), \sum_{c'' \in T}\left(|*, p_x, c''| \times |c'', p_y, *|\right)\right) + 1}, \end{aligned} \tag{5}$$

where p_{pair} is a pattern pair (p_x, p_y); * is any one of all terms or patterns for each position; *c*, *c'*, *c''*, and *c'''* are shared terms; *T* is a set of all terms. (4) and (5) were applied to (1) and (2) of the *Espresso* algorithm.

2.3.3.2. Method Ⅱ

We changed the mutuality equation (*pd*) between the term pair and the pattern in (1) and (2) of the *Espresso* algorithm into an equation for pattern pairs using the combining *pd*s for each pattern of the pattern pair and the weighted shared terms. We defined the mutuality equation (pd_{pair}) between the term pair (x, y) and the pattern pair (p_x, p_y) as:

$$\begin{aligned} pd_{pair}((x,y),p_x,c,p_y) &= \lambda \times pd((x,c),p_x) + (1-\lambda) \times pd((c,y),p_y) \\ &= \frac{P(c,p_y,y) \times pd((x,c),p_x)}{P(x,p_x,c)+P(c,p_y,y)} + \frac{P(x,p_x,c) \times pd((c,y),p_y)}{P(x,p_x,c)+P(c,p_y,y)} \\ &= \frac{|c,p_y,y| \times pd((x,c),p_x)}{|x,p_x,c|+|c,p_y,y|} + \frac{|x,p_x,c| \times pd((c,y),p_y)}{|x,p_x,c|+|c,p_y,y|} \end{aligned} \quad (6)$$

where c is a shared term; λ and 1 - λ are weights using frequencies of term pairs and patterns for each pd. The weight (*Cscore*) of the shared term (c) was defined as:

$$Cscore(x,p_x,c,p_y,y) = P(c \mid x,p_x,p_y,y) = \frac{|x,p_x,c| \times |c,p_y,y|}{\sum_{c' \in T}(|x,p_x,c'| \times |c',p_y,y|)} \quad (7)$$

We calculated pd_{pair}s and *Cscore*s for all shared terms satisfying the pattern pair, and modified (1) and (2) of the *Espresso* algorithm as:

$$r_P(p_x,p_y) = \frac{\sum_{(x,y)\in XY}\left(\sum_{c\in T}\left(Cscore(x,p_x,c,p_y,y) \times pd_{pair}((x,y),p_x,c,p_y)\right) \times r_{XY}(x,y)\right)}{|XY|} \quad (8)$$

$$r_{XY}(x,y) = \frac{\begin{pmatrix} \sum_{p\in P}(pd((x,y),p) \times r_P(p)) \\ + \sum_{(p_x,p_y)\in P}\left(\sum_{c\in T}\left(Cscore(x,p_x,c,p_y,y) \times pd_{pair}((x,y),p_x,c,p_y)\right) \times r_P(p_x,p_y)\right) \end{pmatrix}}{|P|} \quad (9)$$

2.4 Applying modified reliabilities

We applied three cases (Table 2) to *is-a* and *part-of* relation extractions to compare the results. *Case 1* (previous method) used only single patterns, and *Case 2* (Method Ⅰ) and *Case 3* (Method Ⅱ) used single patterns and pattern pairs. For all three cases, we performed 10 iterations, and accumulated 200 term pairs.

Table 2: Equations of three cases for extracting patterns and term pairs.

Case	Equations used			
	Pattern			Term pair
	top 2 single patterns	top single pattern	top pattern pair	top 20 term pairs
1	(1)	not used	not used	(2)
2	not used	(1)	(1) applying (4) and (5)	(2) applying (4) and (5)
3	not used	(1)	(8)	(9)

We evaluated the accuracy of the extracted term pairs, and measured the number of the new term pairs which could not be extracted using single patterns. The accuracy of the term pairs was judged manually by whether the sentence that consisted of the term pair and the typical single pattern reflecting the target relation was natural or not. For example, if "*A* is *B*" was natural, {*A*, *B*} was *is-a* relation, and if "*B* consists of *A*" was natural, {*A*, *B*} was *part-of* relation. We divided the number of the extracted term pairs satisfying the target relation by the number of all the extracted term pairs, and regarded this value as the accuracy. We also counted the new term pairs, and evaluated the accuracy of these.

3 Results

3.1 Overview

We extracted *is-a* relations and *part-of* relations using single patterns and pattern pairs. We prepared 10 seed term pairs for each relation, and extracted 2,409,100 term pairs, 13,538 single patterns, 1,012,334 pattern pairs, and 2,674,684 sets of {term pair, single pattern} from the mass corpus. To generalize patterns partially, we replaced terms in patterns with the unique label (*TR*), and extracted only patterns that appeared more than 20 times in the mass corpus. For each relation extraction, we selected the reliable patterns using the seed term pairs and r_P, and selected the top 20 term pairs using these patterns and r_{XY}. These term pairs were used to extract new patterns. We performed 10 iterations, and evaluated the accumulated 200 term pairs.

3.2 *Is-a* relation extraction

We applied *Case 1*, *Case 2*, and *Case 3* to *is-a* relation extraction. The success rates were 60.5% for *Case 1*, 67.5% for *Case 2*, and 71.5% for *Case 3* (Figure 2). Compared to *Case 1*, the accuracies of *Case 2* and *Case 3* were improved by 7% and 11% respectively. *Case 2* extracted 135 new term pairs which could not be extracted using single patterns (*Case 1*), and these new term pairs achieved 62.22% accuracy. *Case 3* extracted 144 new term pairs, and these term pairs achieved 67.36% accuracy.

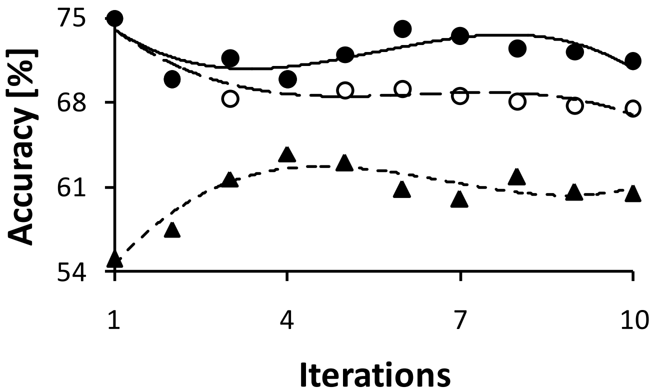

Figure 2: Accuracy vs. iteration in three cases of *is-a* relation extraction. Filled triangles, short-broken line: *Case 1*; Clear circles, long-broken line: *Case2*; Filled circles, solid line: *Case 3*.

In *is-a* relations, *Case 1* could not adequately extract single patterns satisfying the seed term pairs from the mass corpus. This situation produced low accuracy in the first iteration, and the term pairs that had low accuracy subsequently extracted new single patterns that also had low reliability. However, *Case 2* and *Case 3* could use the seed term pairs successfully to extract pattern pairs, and these pattern pairs could also extract reliable term pairs that had high accuracy. Reliable term pairs consistently extracted new reliable single patterns and pattern pairs (Table 3), and this situation produced the high performance. The high accuracy of the new term pairs was also one of reasons for the high performance.

Table 3: Single patterns and pattern pairs in each iteration in *Case 3* of *is-a* relation extraction.

Iteration	Pattern	
	Single pattern	**Pattern pair**
1	*Y* ,/, or/cc a/dt *X*	{*C* on/in his/prp$ *X*, *C* were/vbd on/in the/dt *Y*}
2	*Y* and/cc a/dt *X*	{*C* ,/, a/dt *TR* ,/, or/cc a/dt *X*, *C* ,/, a/dt *Y*}
3	*Y* or/cc a/dt *X*	{*X* of/in the/dt *TR* or/cc a/dt *C*, *Y* or/cc a/dt *TR* of/in the/dt *C*}
4	*X* and/cc one/cd *Y*	{*C* each/dt *X*, *C* a/dt *Y*}
5	*Y* or/cc by/in the/dt *X*	{*C* of/in the/dt *X*, *C* of/in the/dt following/vbg *Y*}
6	*X* ,/, but/cc from/in *Y*	{*C* of/in this/dt *X*, *C* of/in the/dt following/vbg *Y*}
7	*Y* ,/, and/cc a/dt *TR* ,/, and/ cc a/dt *X*	{*X* since/in that/dt *C*, *Y* at/in a/dt *C*}
8	*X* and/cc three/cd *Y*	{*X* after/in that/dt *C*, *Y* at/in a/dt *C*}
9	*X* and/cc six/cd *Y*	{*C* within/in the/dt *X*, *Y* at/in a/dt *C*}
10	*X* and/cc seven/cd *Y*	{*C* within/in a/dt *X*, *Y* at/in a/dt *C*}

3.3 *Part-of* relation extraction

We applied *Case 1*, *Case 2*, and *Case 3* to *part-of* relation extraction. The accuracies were 84% for *Case 1*, 86.5% for *Case 2*, and 88% for *Case 3* (Figure 3). Compared to *Case 1*, the accuracy was 2.5% better for *Case 2* and 4% better for *Case 3*. *Case 2* extracted 97 new term pairs, and these term pairs achieved 86.6% accuracy. *Case 3* extracted 85 new term pairs, and these term pairs achieved 91.76% accuracy.

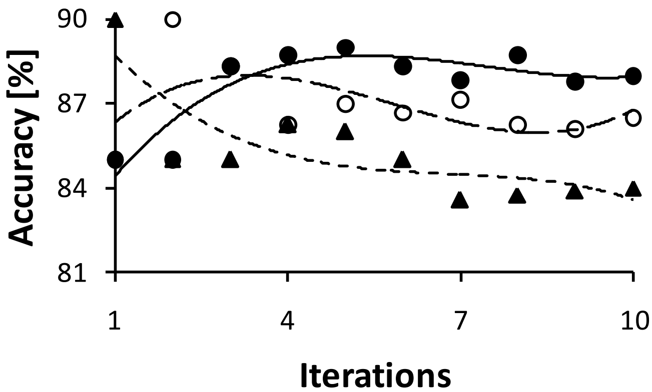

Figure 3: Accuracy vs. iteration in three cases of *part-of* relation extraction. Filled triangles, short-broken line: *Case 1*; Clear circles, long-broken line: *Case2*; Filled circles, solid line: *Case 3*.

In *part-of* relations, *Case 1*, *Case 2* and *Case 3* could successfully extract single patterns or pattern pairs satisfying the seed term pairs from the mass corpus, but *Case 1* could not maintain the high performance of the first iteration. However, *Case 2* and *Case 3* maintained high performances, because these cases continually extracted new reliable single patterns and pattern pairs in each iteration (Table 4), and detected the new term pairs that had high accuracy.

Table 4: Single patterns and pattern pairs in each iteration in *Case 3* of *part-of* relation extraction.

Iteration	Pattern	
	Single pattern	**Pattern pair**
1	*X* of/in a/dt *Y*	{*X* over/in our/prp$ *C*, *C* of/in their/prp$ *TR* 's/pos *Y*}
2	*X* of/in a/dt *TR* 's/pos *Y*	{*X* over/in our/prp$ *C*, *C* of/in the/dt *Y*}
3	*Y* and/cc this/dt *X*	{*C* leading/vbg to/to the/dt *X*, *C* of/in the/dt *Y*}
4	*X* of/in that/dt *Y*	{*X* over/in their/prp$ *C*, *C* of/in their/prp$ *TR* 's/pos *Y*}
5	*X* of/in any/dt *Y*	{*C* of/in the/dt *X*, *C* of/in the/dt *TR* of/in a/dt *Y*}
6	*X* of/in one/cd *Y*	{*C* of/in the/dt *X*, *C* of/in a/dt *Y*}
7	*X* of/in such/jj a/dt *Y*	{*C* -rrb-/-rrb- the/dt *X*, *C* -rrb-/-rrb- a/dt *Y*}
8	*X* as/in a/dt *Y*	{*C* -rrb-/-rrb- the/dt *X*, *C* -rrb-/-rrb- if/in the/dt *Y*}
9	*Y* 's/pos *X*	{*C* -rrb-/-rrb- an/dt *X*, *C* -rrb-/-rrb- a/dt *Y*}
10	*Y* whose/wp$ *X*	{*C* 's/pos *X*, *Y* 's/pos *TR* to/to a/dt *C*}

4 Discussion

In extracting *is-a* relations, the accuracies of our methods *Case 2* and *Case 3* were 67.5% and 71.5% respectively. Compared to the accuracy of the previous method *Case 1* (60.5%), these were improvements of 7% and 11% respectively. In extracting *part-of* relations, the accuracies of our methods *Case 2* and *Case 3* were 86.5% and 88%. Compared to the accuracy of *Case 1* (84%), these were improvements of 2.5% and 4%.

For each relation, *Case 3* had the best performance, and these results mean that the method using pattern pairs was useful in *is-a* and *part-of* relation extractions, and that the method of combining statistical values for each pattern of the pair and the weight of the shared term was more useful than the other methods. Single patterns satisfying a specified term pair could not appear in the corpus, but pattern pairs satisfying the term pair could be extracted from the corpus because each pattern of the pair could appear in the corpus. This characteristic of pattern pairs enabled extraction of reliable term pairs using the reliable pattern pair. These term pairs extracted reliable single patterns and pattern pairs consistently.

For new term pairs which could not be extracted using the previous method that considered only single patterns, we extracted 135 new *is-a* relations (term pairs) and 97 new *part-of* relations using *Case 2*, and 144 new *is-a* relations and 85 new *part-of* relations using *Case 3*. The accuracies of these new term pairs were 62.22%, 86.6%, 67.36%, and 91.76% respectively.

Case 3 extracted many new term pairs that had the best performance, and discoveries of these new term pairs were one of the reasons for the good performance of our methods. We could use the reliable new term pairs to extract single patterns and pattern pairs because pattern pairs could extract reliable pairs of terms in different sentences in the corpus. The reliable pattern pairs detected new term pairs consistently, and the accumulated new term pairs produced the good performance.

Our results were more useful in *is-a* and *part-of* relation extractions than the previous results using single patterns. We will construct various mass corpora, and prepare seed term pairs for various relations, and apply our methods to various other relations.

5 Conclusion

This study proposed a method that extracted term pairs satisfying *is-a* relations or *part-of* relations from a mass corpus using pairs of patterns sharing a term, and compared the result with the previous approach that used single patterns. The proposed method achieved 71.5% accuracy in detecting *is-a* relations, and 88% accuracy in detecting *part-of* relations. Compared to the previous method, these were improvements of 11% and 4% respectively. Furthermore, we extracted 144 new *is-a* relations and 85 new *part-of* relations which could not be extracted using single patterns. These results will be useful in constructing an ontology and a thesaurus because these language knowledge bases consisted mainly of *is-a* relations and *part-of* relations.

References

Berland, M. and E. Charniak. 1999. Finding parts in very large corpora. *Proceedings of Association for Computational Linguistics*, 57-64.

Girju, R., A. Badulescu and D. Moldovan. 2003. Learning semantic constraints for the automatic discovery of part-whole relations. *Proceedings of Human Language Technology / North American Association for Computational Linguistics*, 1-8.

Hearst, M.A. 1992. Automatic acquisition of hyponyms from large text corpora. *Proceedings of the 14th conference on Computational linguistics*, vol. 2, 539-545.

Kavalec, M. and V. Svatek. 2005. A Study on Automated Relation Labelling in Ontology Learning. *Ontology Learning and Population from Text: Methods, Evaluation and Applications*, IOS Press, 44-58.

Maedche, A. and S. Staab. 2002. Measuring similarity between ontologies. *International Conference on Knowledge Engineering and Management*, LNAI 2473, 251-263.

Pantel, P. and D. Ravichandran. 2004. Automatically labeling semantic classes. *Proceedings of Human Language Technology / North American Association for Computational Linguistics*, 321-328.

Pantel, P. and M. Pennacchiotti. 2006. Espresso: Leveraging generic patterns for automatically harvesting semantic relations. *Proceedings of Association for Computational Linguistics*, 113-120.

Ravichandran, D. and E. Hovy. 2002. Learning surface text patterns for a question answering system. *Proceedings of Association for Computational Linguistics*, 41-47.

Yang, X. and J. Su. 2007. Coreference resolution using semantic relatedness information from automatically discovered patterns. *Proceedings of Association for Computational Linguistics*, 528-535.