Combining General Hand-Made and Automatically Constructed Thesauri for Information Retrieval

Rila Mandala, Takenobu Tokunaga and Hozumi Tanaka
Department of Computer Science
Tokyo Institute of Technology
Okayama Meguro Tokyo 152-8552, Japan
{rila, take, tanaka}@cs.titech.ac.jp

Abstract
One of the most intuitive ideas for enhancing the effectiveness of an information retrieval system is to include the use of a thesaurus. WordNet, as a hand-crafted and general-purpose thesaurus, intuitively should also work fine in information retrieval, but unfortunately, experimental results by many researchers have not been promising. Thereby in this paper we investigate why the use of WordNet in information retrieval has not been successful. Based on this analysis we propose a method to combine WordNet with predicate-argument-based and co-occurrence-based automatically constructed thesauri. Experiments using large test collection shows that our method results in a significant improvement of information retrieval performance.

1 Introduction
The task of the information retrieval system is to match the query against the document collection and return relevant documents to the user. Retrieval performance is usually express in terms of recall, the proportion of relevant document retrieved, and precision, the proportion of retrieved document that is relevant. Whereas a perfect retrieval run will have a value of 1.0 for both recall and precision, in practice precision and recall are inversely related.

A critical problem in information retrieval is the case of different words being used to describe the same thing, either in queries or in documents. This require some kind of knowledge base which can equate or relate the terms used in language, i.e., a thesaurus. A thesaurus is a data structure which groups synonymous terms and relates them as either broader or narrower. A thesaurus can be used to expand a query to include all synonymous or related terms. This method has been known as query expansion method [Schutze and Pederson, 1997].

WordNet is currently the most large, hand-crafted, general-purpose, machine-readable, and publically available thesaurus. It is the product of a research project at Princeton University which has attempted to model the lexical knowledge of English [Miller, 1990]. WordNet has been used in numerous natural language processing, such as semantic tagging [Segond et al., 97], word sense disambiguation [Resnik, 1995a], text categorization [Gomez-Hidalgo and Rodriguez, 1997], information extraction [Chai and Biermann, 1997], and so on with considerable success. However, the use of WordNet in information retrieval have not been very successful.

Two sets of experiments using the TREC collection were performed to investigate the effectiveness of using WordNet for query expansion by Voorhees [1994]. The first set used handpicked synsets and the second set extends the expansion strategy to include automatically selecting the starting synsets. When the concepts were chosen manually, her method could improve the retrieval effectiveness for short queries, but failed to improve the retrieval effectiveness for long queries. When the concepts were chosen automatically, none of the expansion methods produced significant improvement as compared with an unexpanded run. She further tried to use WordNet as a tool for word sense disambiguation [Voorhees, 1993] and applied it to text retrieval, but the performance of retrieval was degraded.

Stairmand [1997] used WordNet to investigate the computational analysis of lexical cohesion in text using lexical chain method [Morris and Hirst, 1991]. Because lexical chains are associated with topics, he suggested that information retrieval, where the notion of topic is very pertinent, is a suitable application domain. He concluded that his method only succeed in small-scale evaluation, but a hybrid approach is required to scale-up to real-world information retrieval scenarios.

Smeaton and Berrut [1995] tried to expand the queries of the TREC-4 collection with various strategies of weighting expansion terms, along with manual and automatic word sense disambiguation techniques. Unfortunately all strategies degraded the retrieval performance.

Instead of matching terms in queries and documents, Richardson [1995] used WordNet to compute the semantic distance between concepts or words and then used this term distance to compute the similarity between a query and a document. Although he proposed two methods to compute semantic distances, neither of them increased the retrieval performance.
2 Limitations of WordNet

In this section we analyze why WordNet has failed to improve information retrieval performance. We ran exact-match retrieval against 9 small standard test collections [Fox, 1990] in order to observe this phenomenon. An information retrieval test collection consists of a collection of documents along with a set of test queries. The set of relevant documents for each test query is also given, so that the performance of the information retrieval system can be measured. We expand queries using a combination of synonyms, hypernyms, and hyponyms in WordNet. The results are shown in Table 1.

In Table 1 we show the name of the test collection (Collection), the total number of documents (#Doc) and queries (#Query), and all relevant documents for all queries (#Rel) in that collection. For each document collection, we indicate the total number of relevant documents retrieved (Ret-rel), the recall (Ret-rel/#Rel), the total number of documents retrieved (Ret-docs), and the precision (Ret-docs/#Rel) for each of no expansion (Base), expansion with synonyms (Exp. I), expansion with hypernyms (Exp. II), expansion with synonyms and hypernyms (Exp. III), and expansion with synonyms, hypernyms, and hyponyms (Exp. IV).

From the results in Table 1, we can conclude that query expansion can increase recall performance but unfortunately degrades precision performance. We thus turned to investigation of why all the relevant documents could not be retrieved with the query expansion method above. Some of the reasons are stated below:

- Two terms that seem to be interrelated have different parts of speech in WordNet. This is the case between *stochastic* (adjective) and *statistic* (noun). Since words in WordNet are grouped on the basis of part of speech in WordNet, it is not possible to find a relationship between terms with different parts of speech.
- Most of relationships between two terms are not found in WordNet. For example how do we know that Sumitomo Bank is a Japanese company?
- Some terms are not included in WordNet (proper name, etc).

To overcome all the above problems, we propose a method to enrich WordNet with an automatically constructed thesaurus. The idea underlying this method is that an automatically constructed thesaurus could complement the drawbacks of WordNet. For example, as we stated earlier, proper names and their interrelations among them are not found in WordNet, but if proper names and other terms have some strong relationship, they often co-occur in the document, so that their relationship may be modeled by an automatically constructed thesaurus.

Polysemous words degrade the precision of information retrieval since all senses of the original query term are considered for expansion. To overcome the problem of polysemous words, we apply a restriction in that queries are expanded by adding those terms that are most similar to the entirety of query terms, rather than selecting terms that are similar to a single term in the query.

3 Method

In this section, we first describe the construction method for each type of thesaurus utilized in this research, and then describe a term weighting method using similarity measure based on these thesauri.

3.1 WordNet

In WordNet, words are organized into taxonomies where each node is a set of synonyms (a synset) representing a single sense. There are 4 different taxonomies based on parts of speech and also there are many relationships defined within it [Fellbaum, 1998]. In this experiment we use only noun taxonomy with hypernym/hyponym (or the is-a relation), which relates more general and more specific senses.

The similarity between word \( w_1 \) and \( w_2 \) is defined as a shortest path from each sense of \( w_1 \) to each sense of \( w_2 \), as below [Resnik, 1995b]:

\[
\text{sim}(w_1, w_2) = \max\left(-\log\left(\frac{N_p}{2D}\right)\right)
\]

where \( N_p \) is the number of nodes in path \( p \) from \( w_1 \) to \( w_2 \) and \( D \) is the maximum depth of the taxonomy.

3.2 Co-occurrence-based Thesaurus

The general idea underlying the use of term co-occurrence data for thesaurus construction is that words that tend to occur together in documents are likely to have similar, or related, meanings [Qu and Frei, 1993]. Co-occurrence data thus provides a statistical method for automatically identifying semantic relationships that are normally contained in a hand-made thesaurus. Suppose two words \( A \) and \( B \) occur \( f_a \) and \( f_b \) times, respectively, and co-occur \( f_c \) times, then the similarity between \( A \) and \( B \) can be calculated using a similarity coefficient such as the Tanimoto Coefficient:

\[
\frac{f_c}{f_a + f_b - f_c}
\]

3.3 Predicate-Argument-based Thesaurus

In contrast with the previous section, this method attempts to construct a thesaurus according to predicate-argument structures [Hindle, 1990; Grafenstette, 1994; Ruge, 1992]. The use of this method for thesaurus construction is based on the idea that there are restrictions on what words can appear in certain environments, and in particular, what words can be arguments of a certain predicate. For example, a *dog may walk, bite*, but can not *fly*. Each noun may therefore be characterized according to the verbs or adjectives that it occurs with.
Table 1: Term Expansion Experiment Results using WordNet

<table>
<thead>
<tr>
<th>Collection</th>
<th>#Doc</th>
<th>#Query</th>
<th>#Rel</th>
<th>Rel-ret</th>
<th>Exp. I</th>
<th>Exp. II</th>
<th>Exp. III</th>
<th>Exp. IV</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADI</td>
<td>82</td>
<td>35</td>
<td>170</td>
<td>157</td>
<td>159</td>
<td>166</td>
<td>169</td>
<td>169</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Recal</td>
<td>0.9235</td>
<td>0.9353</td>
<td>0.9765</td>
<td>0.9941</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Ret-docs</td>
<td>2.063</td>
<td>2.2952</td>
<td>2.4227</td>
<td>2.737</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Precision</td>
<td>0.0761</td>
<td>0.0693</td>
<td>0.0653</td>
<td>0.0617</td>
</tr>
<tr>
<td>CACM</td>
<td>3204</td>
<td>64</td>
<td>796</td>
<td>738</td>
<td>756</td>
<td>766</td>
<td>773</td>
<td>773</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Recal</td>
<td>0.9271</td>
<td>0.9497</td>
<td>0.9623</td>
<td>0.9711</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Ret-docs</td>
<td>67.950</td>
<td>86.552</td>
<td>101.654</td>
<td>109.391</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Precision</td>
<td>0.0109</td>
<td>0.0087</td>
<td>0.0076</td>
<td>0.0070</td>
</tr>
<tr>
<td>CISI</td>
<td>1460</td>
<td>112</td>
<td>3114</td>
<td>2.952</td>
<td>3.015</td>
<td>3.076</td>
<td>3.104</td>
<td>3.106</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Recal</td>
<td>0.9479</td>
<td>0.9682</td>
<td>0.9878</td>
<td>0.9968</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Ret-docs</td>
<td>87.895</td>
<td>98.844</td>
<td>106.275</td>
<td>108.970</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Precision</td>
<td>0.0336</td>
<td>0.0305</td>
<td>0.0289</td>
<td>0.0284</td>
</tr>
<tr>
<td>CRAN</td>
<td>1398</td>
<td>225</td>
<td>1838</td>
<td>1.769</td>
<td>1.801</td>
<td>1.823</td>
<td>1.815</td>
<td>1.827</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Recal</td>
<td>0.9625</td>
<td>0.9799</td>
<td>0.9918</td>
<td>0.9875</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Ret-docs</td>
<td>199.469</td>
<td>247.212</td>
<td>284.026</td>
<td>287.028</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Precision</td>
<td>0.0089</td>
<td>0.0073</td>
<td>0.0064</td>
<td>0.0063</td>
</tr>
<tr>
<td>INSPEC</td>
<td>12684</td>
<td>84</td>
<td>2543</td>
<td>2.508</td>
<td>2.531</td>
<td>2.538</td>
<td>2.536</td>
<td>2.542</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Recal</td>
<td>0.9862</td>
<td>0.9953</td>
<td>0.9980</td>
<td>0.9972</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Ret-docs</td>
<td>564.809</td>
<td>735.931</td>
<td>852.056</td>
<td>869.364</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Precision</td>
<td>0.0044</td>
<td>0.0034</td>
<td>0.0030</td>
<td>0.0029</td>
</tr>
<tr>
<td>LISA</td>
<td>6004</td>
<td>35</td>
<td>339</td>
<td>339</td>
<td>339</td>
<td>339</td>
<td>339</td>
<td>339</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Recal</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Ret-docs</td>
<td>148.547</td>
<td>171.808</td>
<td>184.101</td>
<td>188.289</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Precision</td>
<td>0.0023</td>
<td>0.0020</td>
<td>0.0018</td>
<td>0.0018</td>
</tr>
<tr>
<td>MED</td>
<td>1033</td>
<td>30</td>
<td>696</td>
<td>689</td>
<td>662</td>
<td>670</td>
<td>671</td>
<td>673</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Recal</td>
<td>0.9181</td>
<td>0.9511</td>
<td>0.9626</td>
<td>0.9640</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Ret-docs</td>
<td>12.021</td>
<td>16.758</td>
<td>22.316</td>
<td>22.856</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Precision</td>
<td>0.0532</td>
<td>0.0395</td>
<td>0.0300</td>
<td>0.0293</td>
</tr>
<tr>
<td>NPL</td>
<td>11429</td>
<td>100</td>
<td>2083</td>
<td>2.061</td>
<td>2.071</td>
<td>2.073</td>
<td>2.072</td>
<td>2.074</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Recal</td>
<td>0.9894</td>
<td>0.9942</td>
<td>0.9952</td>
<td>0.9942</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Ret-docs</td>
<td>267.158</td>
<td>395.280</td>
<td>539.048</td>
<td>577.033</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Precision</td>
<td>0.0077</td>
<td>0.0052</td>
<td>0.0038</td>
<td>0.0036</td>
</tr>
<tr>
<td>TIME</td>
<td>423</td>
<td>24</td>
<td>324</td>
<td>324</td>
<td>324</td>
<td>324</td>
<td>324</td>
<td>324</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Recal</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Ret-docs</td>
<td>23.014</td>
<td>29.912</td>
<td>33.650</td>
<td>32.696</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Precision</td>
<td>0.0141</td>
<td>0.0108</td>
<td>0.0096</td>
<td>0.0095</td>
</tr>
</tbody>
</table>

Nouns may then be grouped according to the extent to which they appear in similar constructions.

First, all the documents are parsed using the Apple Pie Parser, which is a probabilistic chart parser developed by Sekine and Grishman [1995]. Then the following syntactic structures are extracted:

- Subject-Verb
- Verb-Object
- Adjective-Noun

Each noun has a set of verbs and adjective that it occurs with, and for each such relationship, a Tanimoto coefficient value is calculated.

- $C_{sub}(v_i, n_j) = \frac{f_{sub}(v_i, n_j)}{f_{sub}(v_i)+f_{sub}(n_j)-f_{sub}(v_i, n_j)}$

where $f_{sub}(v_i, n_j)$ is the frequency of noun $n_j$ occurring as the subject of verb $v_i$, $f_{sub}(n_j)$ is the frequency of the noun $n_j$ occurring as subject of any verb, and $f(v_i)$ is the frequency of the verb $v_i$

- $C_{obj}(v_i, n_j) = \frac{f_{obj}(v_i, n_j)}{f_{obj}(v_i)+f_{obj}(n_j)-f_{obj}(v_i, n_j)}$

where $f_{obj}(v_i, n_j)$ is the frequency of noun $n_j$ occurring as the object of verb $v_i$, $f_{obj}(n_j)$ is the frequency of the noun $n_j$ occurring as object of any verb, and $f(v_i)$ is the frequency of the verb $v_i$

- $C_{adj}(a_i, n_j) = \frac{f_{adj}(a_i, n_j)}{f(a_i)+f_{adj}(a_i, n_j)-f_{adj}(a_i, n_j)}$

where $f_{adj}(a_i, n_j)$ is the frequency of noun $n_j$ occurring as argument of adjective $a_i$, $f_{adj}(n_j)$ is the frequency of the noun $n_j$ occurring as argument of any adjective, and $f(a_i)$ is the frequency of the adjective $a_i$

We define the similarity of two nouns with respect to one predicate as the minimum of each Tanimoto coefficient with respect to that predicate, i.e.,

- $SIM_{sub}(v_i, n_j, n_k) = \min\{C_{sub}(v_i, n_j), C_{sub}(v_i, n_k)\}$
- $SIM_{obj}(v_i, n_j, n_k) = \min\{C_{obj}(v_i, n_j), C_{obj}(v_i, n_k)\}$
- $SIM_{adj}(a_i, n_j, n_k) = \min\{C_{adj}(a_i, n_j), C_{adj}(a_i, n_k)\}$
3.4 Expansion Term Weighting Method

A query \( q \) is represented by a vector \( \overrightarrow{q} = (q_1, q_2, ..., q_n) \), where the \( q_i \)'s are the weights of the search terms \( t_i \) contained in query \( q \).

The similarity between a query \( q \) and a term \( t_j \) can be defined as follows [Qiu and Frei, 1993]:

\[
sim_{qt}(q, t_j) = \sum_{t_i \in q} q_i \cdot \sigma_{mt}(t_i, t_j)
\]

Where the value of \( \sigma_{mt}(t_i, t_j) \) can be defined as the average of the similarity values in the three types of thesaurus.

With respect to the query \( q \), all the terms in the collection can now be ranked according to their \( \sigma_{qt} \). Expansion terms are terms \( t_j \) with high \( \sigma_{qt}(q, t_j) \).

The weight \( w(q, t_j) \) of an expansion term \( t_j \) is defined as a function of \( \sigma_{qt}(q, t_j) \):

\[
weight(q, t_j) = \frac{\sigma_{qt}(q, t_j)}{\sum_{t_i \in q} q_i}
\]

where \( 0 \leq weight(q, t_j) \leq 1 \).

An expansion term gets a weight of 1 if its similarity to all the terms in the query is 1. Expansion terms with similarity 0 to all the terms in the query get a weight of 0. The weight of an expansion term depends both on the entire retrieval query and on the similarity between the terms. The weight of an expansion term can be interpreted mathematically as the weighted mean of the similarities between the term \( t_j \) and all the query terms. The weight of the original query terms are the weighting factors of those similarities.

Therefore the query \( q \) is expanded by adding the following query:

\[
\overrightarrow{q}^e = (a_1, a_2, ..., a_r)
\]

where \( a_j \) is equal to \( weight(q, t_j) \) if \( t_j \) belongs to the top \( r \) ranked terms. Otherwise \( a_j \) is equal to 0.

The resulting expanded query is:

\[
\overrightarrow{q}_{\text{expanded}} = \overrightarrow{q} \circ \overrightarrow{q}^e
\]

where the \( \circ \) is defined as the concatenation operator.

The method above can accommodate the polysemous word problem, because an expansion term which is taken from a different sense to the original query term is given very low weight.

4 Experimental Results

In order to evaluate the effectiveness of the proposed method in the previous section we conducted experiments using the TREC-7 information retrieval test collection [Voorhees and Harman, to appear 1999]. TREC-7 documents consists of the Financial Times (FT), Federal Register (FR94), Foreign Broadcast Information Service (FBIS) and the LA Times. Table 2 gives its document statistics, Table 3 gives topic statistics, and Table 4 is one example out of 50 topics. As a baseline we used SMART [Salton, 1971] without expansion. SMART is an information retrieval engine based on the vector space model in which term weights are calculated based on term frequency, inverse document frequency and document length normalization. The results are shown in Table 5. This table shows the average of non-interpolated recall-precision for each of baseline, expansion using only WordNet, expansion using only predicate-argument-based thesaurus, expansion using only co-occurrence-based thesaurus, and expansion using all of them. For each method we give the percentage of improvement over the baseline. It is shown that the performance using the combined thesauri for query expansion is better than both SMART and using just one type of thesaurus.
Table 4: Average non-interpolated precision for expansion using combined thesaurus and expansion using only one type of thesaurus.

<table>
<thead>
<tr>
<th>Topic Type</th>
<th>Base</th>
<th>Expanded with</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>WordNet only</td>
<td>Pred-Arg only</td>
</tr>
<tr>
<td>Title</td>
<td>0.117 (+3.6%)</td>
<td>0.131 (+5.2%)</td>
</tr>
<tr>
<td>Desc</td>
<td>0.142 (+5.5%)</td>
<td>0.162 (+7.3%)</td>
</tr>
<tr>
<td>All</td>
<td>0.197 (+7.5%)</td>
<td>0.212 (+10.2%)</td>
</tr>
</tbody>
</table>

5 Discussion
In this section we discuss why our method of using WordNet is able to improve the performance of information retrieval. The important points of our method are:

- the coverage of WordNet is broadened
- weighting method

The three types of thesaurus we used have different characteristics. Automatically constructed thesauri add not only new terms but also new relationships not found in WordNet. If two terms often co-occur together in a document then those two terms are likely bear some relationship. Why not only use the automatically constructed thesauri? The answer to this is that some relationships may be missing in the automatically constructed thesauri [Grafenstette, 1994]. For example, consider the words tumor and tumour. These words certainly share the same context, but would never appear in the same document, at least not with a frequency recognized by a co-occurrence-based method. In general, different words used to describe similar concepts may never be used in the same document, and are thus missed by the co-occurrence methods. However their relationship may be found in the WordNet thesaurus.

The second point is our weighting method. As already mentioned before, most attempts at automatically expanding queries by means of WordNet have failed to improve retrieval effectiveness. The opposite has often been true: expanded queries were less effective than the original queries. Beside the "incomplete" nature of WordNet, we believe that a further problem, the weighting of expansion terms, has not been solved. All weighting methods described in the past researches of query expansion using WordNet have been based on "trial and error" or ad-hoc methods. That is, they have no underlying justification.

The advantages of our weighting method are:

- the weight of each expansion term considers the similarity of that term with all terms in the original query, rather than to just one or some query terms.
- the weight of the expansion term accommodates the polysemous word problem.

This method can accommodate the polysemous word problem, because an expansion term taken from a different sense to the original query term sense is given very low weight. The reason for this is that, the weighting method depends on all query terms and all of the thesaurus. For example, the word bank has many senses in WordNet. Two such senses are the financial institution and the river edge senses. In a document collection relating to financial banks, the river sense of bank will generally not be found in the co-occurrence-based thesaurus because of a lack of articles talking about rivers. Even though (with small possibility) there may be some documents in the collection talking about rivers, if the query contained the finance sense of bank then the other terms in the query would also concerned with finance and not rivers. Thus rivers would only have a relationship with the bank term and there would be no relationships with other terms in the original query, resulting in a low weight. Since our weighting method depends on both query in its entirety and similarity in the three thesauri, the wrong sense expansion terms are given very low weight.

We also experimented this method using other similarity coefficient method and Roget thesaurus, and found significant improvement of retrieval performance although the contribution of Roget thesaurus is very limited [Mandala et al., to appear 1999].

6 Conclusion
This paper analyzed why the use of WordNet, a large and hand-made publically available thesaurus was not so succesful in improving the retrieval effectiveness in information retrieval applications. We found that the main reason is that most relationships between terms are not found in WordNet, and some terms, such as proper names, are not included in WordNet. To overcome this problem we proposed a method to enrich the WordNet with automatically constructed thesauri.

Another problem in query expansion is that of polysemous words. Instead of using a word sense disambiguation method to select the appropriate sense of each word, we overcome this problem with a weighting method. Experiments proved that our method of using WordNet in query expansion could improve information retrieval effectiveness.

In the future, we will use anaphora resolution to accurately determine the nature of relationships involving proper names. We will also investigate the effect of using different similarity coefficient method to build thesauri to the retrieval performance.

7 Acknowledgements
The authors would like to thank Mr. Timothy Baldwin (TIT, Japan) and three anonymous reviewers for useful comments on the earlier version of this paper. We also thank to Dr. Chris Buckley (SabiR Research) for the SMART support, and Dr. Satoshi Sekine (New York University) for providing the Apple Pie Parser program. This research is partially supported by the JSPS project number JSPS-RFTF96P00502.
References


