Query Expansion Using Domain-Adapted, Weighted Thesaurus in an Extended Boolean Model

Oh-Woog Kwon, Myoung-Cheol Kim, Key-Sun Choi
Center for Artificial Intelligence Research
Department of Computer Science
Korea Advanced Institute of Science and Technology
373-1, Kusung-dong, Yusung-gu, Taejon, 305-701, Korea
email : {ohwoog, mckim, kschoi}@csking.kaist.ac.kr

Abstract

In this paper, we address three important issues with query expansion using a thesaurus: how to give weights to the terms in expanded queries, how to select additional search terms in the thesaurus, and how to enrich the terms in the manual thesaurus(namely, thesaurus reconstruction). To weight the terms in expanded queries, we construct the weighted thesaurus that has a similarity value between the terms in the thesaurus, using statistical co-occurrence in a corpus. To enrich the terms in the manual thesaurus, domain dependent terms which occur in a corpus are inserted into the weighted thesaurus using the co-occurrence information. In this paper, the reconstructed thesaurus with weights is defined as a domain-adapted, weighted thesaurus. Then we explain query expansion using the domain-adapted, weighted thesaurus in an extended Boolean retrieval model. To select additional search terms during query expansion, our model use semi-automatic query expansion and a restriction method. In the experiments, our system had almost twice the recall of the boolean retrieval system not using the thesaurus or the query expansion retrieval system using the original thesaurus. And also, the precision of our system was almost the same precision as the other systems.

1 Introduction

Generally speaking, the most significant purpose of Information Retrieval(IR) is to satisfy user's information needs. The user's information needs are expressed by a combination of terms which is called a query. Users think that terms used in a query represent their information needs. But in the IR system, they could not select terms that completely represent the information needs, because users often input queries containing terms that do not match the terms used to index the majority of the relevant documents, and almost always some of the unretrieved relevant documents are indexed by a different set of terms than those in the query[Harmon92].

To solve this problem, many researchers have concentrated on query expansion. Query expansion is the method that expands user's query by adding terms related to each term in the query. The related terms are selected terms in the thesaurus or in a set of co-occurrence terms[Peat91][Sme83]. In query expansion, the number of retrieved documents is proportional to the number of additional search terms. Because the additional search terms are closely related to the terms of user's query, some unretrieved relevant documents are retrieved. Hence, query expansion enhances recall of IR systems[Miy90][Oga91]. But, the enhancement of recall can generally bring about the decline of precision. Both recall and precision of the systems using query expansion are dependent on the selection of additional search terms.

The purpose of this paper is to show how to enhance both recall and precision using query expansion with carefully selected terms in a thesaurus. Three important issues with query expansion are as follows:

1. How to give weights to the terms in expanded queries.
2. How to expand terms of user's query that are not in a thesaurus.
3. How to select additional search terms in a thesaurus.

For the first issue, we suggest that a thesaurus have a similarity value between each term in the thesaurus. The similarity is measured by Mutual Information(MI) that is a statistical value of co-occurrence terms in a corpus. We call such a thesaurus "Weighted Thesaurus". We use the weights of terms in the weighted thesaurus during query expansion and document ranking.

When the terms in user's query are not in a thesaurus, we can not process query expansion. This situation occurs frequently in IR systems, because the well-defined thesaurus for the specific domain is not available in most cases. A thesaurus should be reconstructed by adding terms in that domain with high values of MI. We call the thesaurus "Domain-Adapted, Weighted thesaurus". Because the similarity values between terms in the thesaurus are changed according to the domain of documents, such thesaurus is flexible to the domain.

For the final issue, we restrict additional search terms to some level of a domain adapted, weighted thesaurus, and we use semi-automatic query expansion.

In section 2, we describe a method of constructing a weighted thesaurus based on a manual thesaurus and MI. Section 3 describes a method of building a domain-adapted, weighted thesaurus to enrich domain dependent terms. A ranking method in expanded query is explained in section 4. In section 5, some results of experiments carried out with 1,000 sample documents are presented.

CIKM '94- 11/94 Gaitherburg MD USA

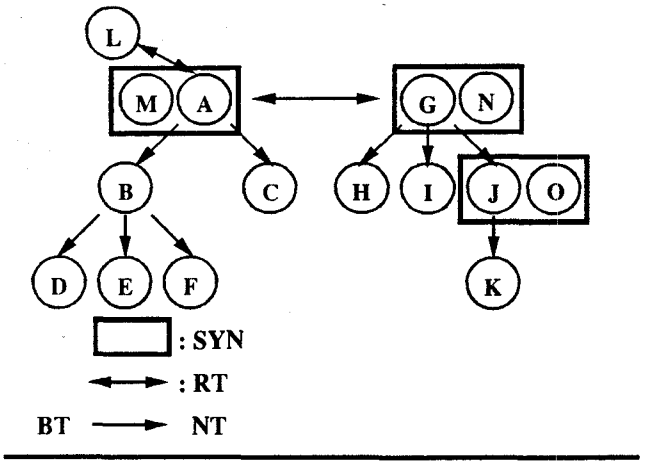

Figure 1: Structure of General Thesaurus

2 Weighted Thesaurus

Generally, thesauri have different sets of relationships according to their purpose. For our query expansion, a thesaurus has four kinds of related terms as follows[Harmon92]:

- BT(Broader Term) : provides a more general term.
- NT(Narrower Term) : suggests a more specific term.
- SYN(Synonymous Term) : signifies a synonymous term.
- RT(Related Term) : indicates a term that has various relation such as part-whole or object-property.

In the thesaurus, the relationships such as NT and BT have a hierarchical structure. Hence this structure is presented by a tree. A node of the tree can also have links to SYN terms and RTs (see Figure 1).

Generally, a manual thesaurus does not have any link values which means similarity or concept distances among terms. The similarity between two terms in the thesaurus is reciprocal to the distance. In some IR system uses such a manual thesaurus for query expansion, the distance would be measured by the number of links between two terms in the thesaurus. For example, the distance between term A and B, A and C, B and D, etc. are of the same value 1 in Figure 1. Namely, the same relationship between terms is considered to have the same distance value. But such a measurement can not be true in the real world and in our intuitivity.

In this viewpoint, we formulate a new similarity measure of terms in the thesaurus. For this purpose, we use MI that is a co-occurrence information of two terms in a corpus[Kim92]. MI is a measure of the interdependence of two signals in a message[Fano61]. This MI is a function of the probabilities of two events:

$$MI(x, y) = \log \frac{P_{x,y}(x, y)}{P_x(x)P_y(y)}$$

Consider these events not as signals but as terms in corpus. Then an estimate of MI of two terms, is :

$$MI(x, y) = \log \frac{\frac{\#\ xy\ within\ some\ window\ in\ corpus}{total\ \#\ of\ terms\ in\ corpus}}{(\frac{\#\ x}{total\ \#})(\frac{\#\ y}{total\ \#})}$$

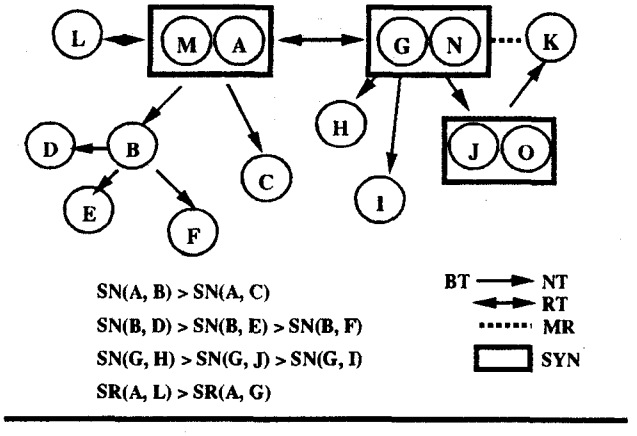

Figure 2: Structure of Weighted Thesaurus

We use this MI for estimating the similarity of two terms A and B in the thesaurus. And the similarity formula of each relation are as follows: (SR, SN and SS are a function of similarity between two terms for RT, NT and BT respectively.)

- **RT** :
  $SR(A, B) = nf(\log_2 \frac{P(A, B)}{P(A)P(B)})$

  Where
  $P(A, B) = \frac{freq(A, B)}{N}$
  $P(A) = \frac{freq(A)}{N}$
  N = *total number of words*
  nf : *normalize function* [0, 1]

- **NT** : $A \supset B$
  $SN(A, B) = nf(\log_2 \frac{P(A, B)}{P(B)})$

- **SYN** :
  $SS(A, B) \approx 1$

The value between two terms is attached to the link between the terms in Figure 1. Figure 2 shows the thesaurus having similarity values. The similarity estimated by MI is inversely proportional to the concept distance. Hence, the longer the length of a link in Figure 2 is, the lower the similarity between those terms connected to the link is. A weighted thesaurus is useful for changing user's query to a weighted query during query expansion.

In Figure 2, we introduce a new relationship MR(Mutual Relationship). MR describes a relationship between a term and other term having no link directly connected to the term in Figure 1. MR occurs in the situation that the similarity of two terms is higher than the similarity manually assigned. So MR is useful in choosing additional search terms and in correcting errors of a manual thesaurus.

3 Domain-Adapted, Weighted Thesaurus

Construction of a manual thesaurus is very difficult and takes a long time. And it must be changed and maintained periodically to incorporate new and modified terms. Also, relationships must be updated, because as time goes by, the meaning of term is changed and a lot of new terms emerges. Updates are slow and require many people who will reconstruct the thesaurus[Harmon92]. Thesaurus reconstruction

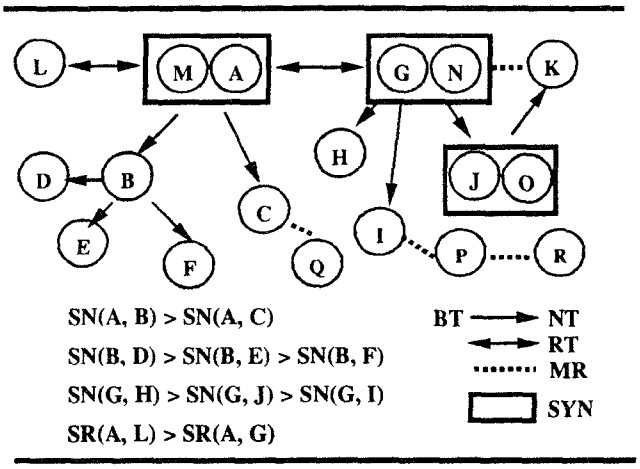

Figure 3: Structure of Domain-Adapted, Weighted Thesaurus

is frequently required in changing IR applications, because free terms almost always occur in user's queries and in documents of new doamin. To solve the problem, we provide an automatic insertion mechanism of new terms into the thesaurus.

We reconstruct a weighted thesaurus by adding domain dependent terms that have high values of co-occurrence in a corpus. We call this new thesaurus "Domain-Adapted, Weighted Thesaurus". The relationship between additional terms, and between an additional term and a term in the thesaurus is defined by MR(Mutual Relation). In section 2, MR are used to define new relationships that MI brings about. The formula for RT given in section 2 also works for MR. For example, if free terms P, Q and R were inserted in Figure 2, the structure in Figure 2 would be changed to Figure 3.

The following algorithm assigned the similarity value between two terms in the manual thesaurus and additional domain dependent terms:

Algorithm for Domain-Adapted, Weighted Thesaurus.

T is a set of terms in the manual thesaurus.
D is a set of additional domain dependent terms.
U = **T** ∪ **D**.
C is corpus.
$SV(a, b)$ is the similarity value between term a and b.
α is threshold value of **RT**.
β is threshold value of **NT**.
γ is threshold value of **SYS**.
δ is threshold value of **MR**.

```
for each term a in C
    if a ∈ U then
      freq(a) = freq(a) + 1 ;
    if a and b ∈U within some window then
      freq(a, b) = freq(a, b) + 1 ;
end for;

for each pair (a, b) such as a and b in U
    if a and b in T then
      if the relation between a and b is RT then
        if SR(a, b) > α then
          SV(a, b) = SR(a, b);
        else SV(a, b) = α ;
      else if the relation from a to b is NT then
            if SN(a, b) > β then
              SV(a, b) = SN(a, b);
            else SV(a, b) = β;
      else if the relation between a and b is SYN then
            if SS(a, b) > γ then
              SV(a, b) = SS(a, b);
            else SV(a, b) = γ;
      else
            if SR(a, b) > δ then
              SV(a, b) = SR(a, b);
              Assigne MR link between a and b;
    /* So far, the procedure for Weighted Thesaurus */
    else /* For Domain-Adapted, Weighted Thesaurus */
      if SR(a, b) > δ then
        SV(a, b) = SR(a, b);
        Assign MR link between a and b;
end for;
```

In the above algorithm, each threshold values(except MR) work for the default similarity value of two terms in the manual thesaurus.

So far we have explained the Domain-Adapted, WeIghted Thesaurus(DAWIT). Characteristics of DAWIT are as follows:

1. The structure of DAWIT can be flexibly adapted to document domain.
2. DAWIT has similarity values between the terms.
3. DAWIT is a graph structure.
4. DAWIT can correct errors of the original thesaurus.
5. During query expansion, the user's query can be changed to corresponding weighted query using DAWIT.

We addressed the new relation MR. This relation resembles RT and links two node(term)s that have no direct relation in the thesaurus. So, MR causes one node to have many domain dependent nodes and directly to link nodes that have almost the closeness of concept in the domain. This MR is useful for query expansion. In query expansion, our system expands query with terms that directly link to terms of query in DAWIT, because in DAWIT, one node have enough many children and those children are closely related to the terms in query. But MR relationship should be carefully treated, because it is heavily dependent on training corpus.

4 Query Expansion using DAWIT

Our system is based on an extended Boolean model[Lee93]. Original query Q of a user has no weight and is expressed by (1).

$$Q = T_i(\odot T_j)^* \qquad (1)$$

$\odot$: *Boolean Operator* (AND, OR, NOT)

If a term T is in DAWIT, T has children as follows.

$$RT,\ MR\ :\ T_r(S_{TT_r}\ :\ Similarity(Weight)\ of\ T\ and\ T_r)$$
$$SYN\ :\ T_s(S_{TT_s}\ :\ Similarity(Weight)\ of\ T\ and\ T_s)$$
$$NT\ :\ T_n(S_{TT_n}\ :\ Similarity(Weight)\ of\ T\ and\ T_n)$$

The term T is expanded as (2) using its children and weights of its children.

$$T' = (T,\ 1)\ OR\ (T_r,\ S_{TT_r})\ OR\ (T_s,\ S_{TT_s})\ OR\ (T_n,\ S_{TT_n}) \quad (2)$$

If the term T is not in DAWIT, the term T is expanded as (3).

$$T' = (T,\ 1) \quad (3)$$

Because as we expressed in section 3, MR describes the relevant terms to the term in user's query, when a term is expanded, we restrict additional search terms to its children in DAWIT.

After all terms in an original query are processed by term expansion procedure, an expanded query Q' is expressed as (4). The expanded query has weights of terms.

$$Q' = T_i'(\odot T_j')_* \quad (4)$$

Now, we explain documents ranking based on the extended boolean model. If a term T is weighted as tw during indexing and as qw in an expanded query, a document-query similarity T_W is expressed as (5). However, if the term T is a operand of NOT operator, T_W is expressed as (6) and NOT operator is changed to AND operator.

$$T_W = tw \times qw \quad (5)$$

$$T_W = (1 - tw) \times qw \quad (6)$$

Expressions in the extended Boolean model for AND and OR operators are as follows:

$$(T_i, W_i)\ AND\ (T_j, W_j) = 0.8 \times \min(W_i, W_j) + 0.2 \times \frac{W_i + W_j}{2} \quad (7)$$

$$(T_i, W_i)\ OR\ (T_j, W_j) = 0.8 \times \max(W_i, W_j) + 0.2 \times \frac{W_i + W_j}{2} \quad (8)$$

The expression $\frac{W_i+W_j}{2}$ in (7) and (8) plays a role of positive compensation to min and max operators[Lee93].

5 Test and Evaluation Results

We have proposed an Retrieval model for query expansion using DAWIT. The diagram of our model is described in Figure 4. Our model has been implemented in C under the UNIX environment to process Korean language.

For our experiments, we used the test collection as follows:

- Manual Thesaurus : has 5,808 terms for general area[KT93] and is constructed by Ehoa Woman University.
- Documents & Queries[KT94]
  - Documents : 1,000 paper abstractions of KISS(Korea Information Science Society) in computer science.
  - Queries : KT Test Set which is constructed by Korea Telecom Research Center, has 30 queries and the set of relevant documents to each query.

An example of query expansion in our system is as follows[1]

:

[1]Korean terms in examples are translated into English.

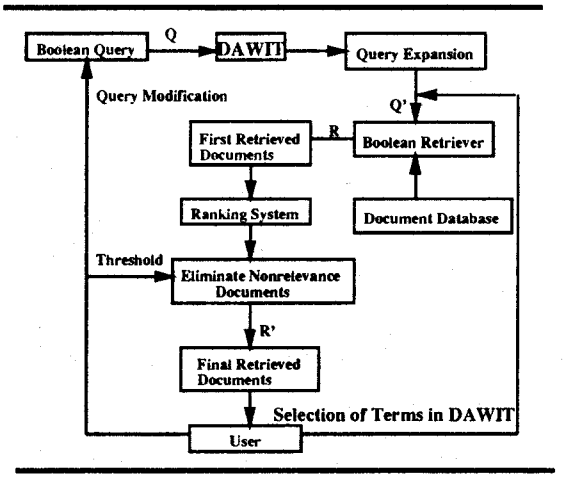

Figure 4: Diagram of Boolean Retrieval Model

- User query : *"Database"* and *"Real-Time"*
- Directly-related terms of *Database* in DAWIT.

Term	Relation	Similarity Value
Relation	MR	0.37
Data Model	MR	0.35
Data Theory	MR	0.44
Second Memory	MR	0.49
Information Management	MR	0.34
Query	RT	0.40
Schema	RT	0.38
⋮	⋮	⋮

- Directly-related terms of *Real-Time* in DAWIT.

Term	Relation	Similarity Value
Shared Memory	MR	0.37
Mamagement Method	MR	0.38
Ring	MR	0.38
Memory Access	MR	0.57
Recovery	MR	0.42
Security	MR	0.42
Scheduling	MR	0.43
Real-Time System	MR	0.43
Operating System	MR	0.38
⋮	⋮	⋮

- Expanded query :
  (*"Database"* or *"Relation"* or *"Data Model"* or *"Schema"* or *"Query"* or ...) and (*"Real-Time"* or *"Real-Time System"* or *"Real-Time Environment"* or *"Recovery"* or ...)

In the above example, query expansion is done semi-automatically. Expanded query is weigthed during retrieval precessing. Figure 5 shows our system that is processed by the above example.

We compared the retrieval effectiveness of three cases. Three cases are as follows:

- CASE I : The Boolean retrieval model not using thesaurus.

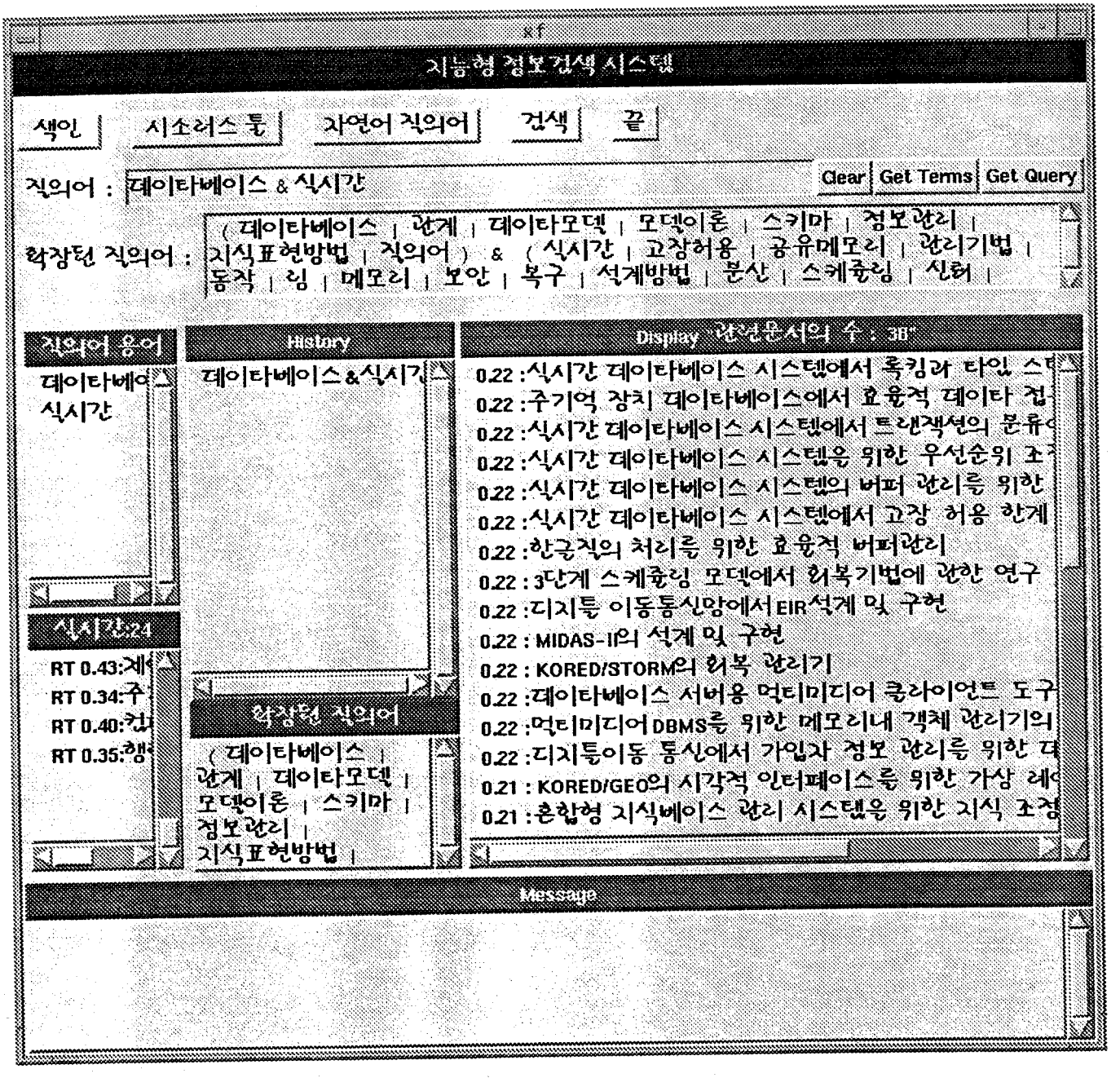

Figure 5: For example, retrieved documents using query system in our system

Table 1: The Results

Case	Measurements	25%	50%	75%	100%
CASE *I*	Avg. Recall	0.059	0.099	0.128	0.172
	Avg. Precision	0.409	0.408	0.387	0.408
CASE *II*	Avg. Recall	0.061	0.107	0.137	0.179
	Avg. Precision	0.443	0.435	0.411	0.430
CASE *III*	Avg. Recall	0.172	0.232	0.278	0.312
	Avg. Precision	0.484	0.428	0.385	0.376

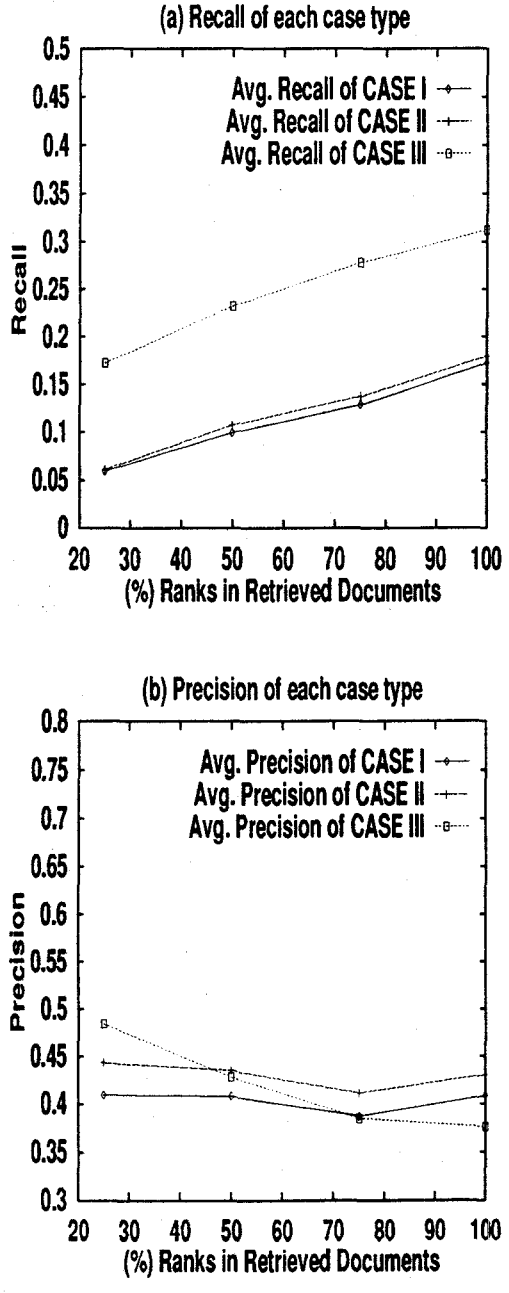

Figure 6: Precision and Recall in each cases

- CASE *II* : Our retrieval model using queries expanded by only weighted thesaurus(WIT).
- CASE *III* : Our retrieval model using queries expanded by DAWIT.

Table 1 and Figure 6 show the results evaluated by the documents which were ranked within 25%, 50%, 75%, and 100% in retrieved documents. In the results of documents within 25% ranks, the recall of case *III* is about three times the recall of case *I* or case *II* and also the precision of case *III* is higher than the precision of case *I* or case *II*. This result confirms our expectation. In the results of all retrieved documents, our model retrieved more relevant documents than other cases. But our model had slightly lower precision than other cases.

Almost relevant documents of other cases were within 25% ranks of our model. It is the reason that the additional search terms in expanded queries increase the similarity between the documents and the queries during document ranking.

The recalls of ten queries in KT Test Set were zero in cases *I* and case *II*, and the recalls of six queries in the set were zero in case *III*. Those zero values decreased the average of recall or precision of all cases. This situation happens in the environment that the terms in the original queries don not occur in the set of documents and the manual thesaurus. If such terms are inserted into the thesaurus and we construct DAWIT using a large set of documents, our system will have better performance.

6 Conclusion

In this paper, we have defined "Weighted Thesaurus(WIT)" and "Domain-Adapted, Weighted Thesaurus(DAWIT)". Query expansion using DAWIT is presented in the extended boolean model. A weighted thesaurus is a thesaurus that has a weight which means the similarity between the terms. The similarity is measured by MI that is a value of co-occurrence in a corpus. If the thesaurus does not include all terms in user's queries, we must reconstruct the thesaurus by adding new terms expected to occur in user's queries. Thesaurus reconstruction is a difficult and long-term task. However, we automatically added new terms that occurred in the corpus, into the weighted thesaurus. The insertion method used MI for determining whether a term was inserted into the weighted thesaurus and weighting a similarity value between terms. Such thesaurus was defined as a "Domain-Adapted, Weighted Thesaurus". In the extended Boolean retrieval model, we expanded user's query using DAWIT. The expanded query had weights of the terms. Such weights were used by ranking algorithm of the extended boolean Retrieval model.

In our experiment, we compared three cases. The first case is the Boolean retrieval model using original user's queries, the second case is query expansion using WIT, and the third case is query expansion using DAWIT. The third case had almost the same precision of other cases and had almost twice recall of other cases. If we construct DAWIT using a large corpus, the performance of our system will be more improved.

In our future research we will concentrate on automatic query expansion to select additional search terms. And we will observe MR structure in DAWIT and construct more reasonable DAWIT.

References

[Salton83] G. Salton, E.A. Fox, and H. Wu, *Extended Boolean Information Retrieval.* Communications of ACM, 26(11), pp. 1022-1036, 1983.

[Salton89] G. Salton, *Automatic Text Processing : The Transformation, Analysis , and Retrieval of Information by Computer.* Addison Wesley, 1989.

[Fano61] Fano, R., *Transmission of Information.* New York, New York: MIT Press, 1961.

[Harmon92] Donna Harman, *Relevance Feedback and Other Query Modification Techniques.* in Information Retrieval : *Data Structures & Algorithms*, ed. William B.Frakes, Ricardo Baeza-Yates, 1992. pp. 241-263. Englewood cliffs, N.J : Prentice Hall.

[Kim90] Kim, Y.H., Kim, J.H., *A Model of Knowledge Based Information Retrieval with Hierarchical-Concept Graph.* Jounal of Documentation, 46(2) pp. 113-136, 1990.

[Kim92] Kim Myung-Chul, Lee Woon-Jae, Choi Key-Sun, Kim Gil-chang, *System for Concept Acquisition for Thesaurus Construction.* Proceedings92 of Hangul and Korean Language Processing, pp 39-49, 1992.

[KT94] Korean Telecom, *Development of Test Set for Evaluation of Automatic Indexer.* Korean Society of Information Management, 1994.5

[KT93] Korea Advanced Institute of Science and Technology, *Intelligent Information Retrieval Environment.* report of Korean Telecom Research Center, 1993. 12

[Lee93] Lee, J.H., Kim, W.Y., Kim, M.H., Lee, Y.J., *On the Envalution of Boolean Operator in the Extended Boolean Retrieval Framework.* ACM-SIGIR '93, pp. 291-297, 1993.

[Peat91] Peat, H.J., Willett, P., *The limitations of term co-occurrence data for query expansion in document retrieval system.* Journal of the *ASIS*, 42(5), pp. 378-383, 1991.

[Rada91] Rada R, Barlow J., *Document Ranking using an Enriched Thesaurus.* Journal of Documentation, 47(3), pp. 240-253, 1991.

[Miy90] S. Miyamoto, *Information Retrieval based on Fuzzy Association* Fuzzy Sets and Systems 38, pp. 191-205, 1990.

[Sme83] Smeaton, A.F., van Rijsbergen, C.J., *The retrieval effects of query expansion on a feedback document retrieval system.* The Computer Journal, 26(3), pp. 239-246, 1983.

[Oga91] Y. Ogawa et al, *A Fuuzy Document Retrieval System Using the Keyword Connection Matrix and a Learning Method.* Fuzzy Sets and Systems 39, pp. 163-179, 1991.