퍼지 및 지능시스템학회 논문지 2001, Vol. 11, No. 7, pp. 659-665

Query Expansion Through Relevance Feedback and Local Context Analysis

Young-cheon kim, You-Mi Moon and Sung-joo Lee

Department of Computer Science, Chosun University

Abstract

Relevance feedback is the most popular query reformulation strategy. in a relevance feedback cycle, the user is presented with a list of the retrieved documents and, after examining them, marks those which are relevant. In practice, only the top 10(or 20) ranked documents need to be examined. The main idea consists of selecting important terms, or expressions, attached to the documents that have been identified as relevant by the user, and of enhancing the importance of these terms in a new query formulation. The expected effect is that the new query will be moved towards the relevant documents and away from the non-relevant ones.

Local analysis techniques are interesting because they take advantage of the local context provided with the query. In this regard, they seem more appropriate than global analysis techniques. In a local strategy, the documents retrieved for a given query q are examined at query time to determine terms for query expansion. This is similar to a relevance feedback cycle but might be done without assistance from the user.

Key words : Relevance feedback, local context analysis, similar, query expansion.

1. Introduction

The Boolean model is a simple retrieval model based on set theory and Boolean algebra[1]. Since the concept of a set is quite intuitive, the Boolean model provides a framework which is easy to grasp by a common user of and IR system. Furthermore, the queries are specified as Boolean expressions which have precise semantics. Given its inherent simplicity and neat formalism, the Boolean model received great attention in past years and was adopted by many of the early commercial bibliographic systems.

Unfortunately, the Boolean model suffers from major drawbacks. First, its retrieval strategy is based on a binary decision criterion without any notion of a grading scale, which prevents good retrieval performance. Thus, the Boolean model is in reality much more a data retrieval model. Second, while Boolean expressions have precise semantics, frequently it is not simple to translate an information need into a Boolean expression. In fact, most users find it difficult and awkward to express their query requests in terms of Boolean expressions.

The vector model recognizes that the use of binary weights is too limiting and proposes a framework in which partial matching is possible.

This is accomplished by assigning non-binary weights to index terms in queries and in documents. These term weights are ultimately used to compute the degree of similarity between each document stored in the system and the user query.

By sorting the retrieved documents in decreasing order of this degree of similarity, the vector model takes into consideration documents which match the query terms only partially. The main resultant effect is that the ranked document answer set is a lot more precise than the document answer set retrieved by the Boolean model.

Relevance feedback is the most popular query reformulation strategy[2]. in a relevance feedback cycle, the user is presented with a list of the retrieved documents and, after examining them, marks those which are relevant. In practice, only the top 10(or 20) ranked documents need to be examined. The main idea consists of selecting important terms, or expressions, attached to the documents that have been identified as relevant by the user, and of enhancing the importance of these terms in a new query formulation. The expected effect is that the new query will be moved towards the relevant documents and away from the non-relevant ones.

Local analysis techniques are interesting because they take advantage of the local context provided with the query. In this regard, they seem more appropriate than global analysis techniques. In a local strategy, the documents retrieved for a given query q are examined at query time to determine terms for query expansion. This is similar to a relevance feedback cycle but might be done without assistance from the user.

The approach is based on the use of noun groups, instead if simple keywords, as document concepts. For query expansion, concepts are selected from the top ranked documents based on their co-occurrence with query terms. However, instead of documents, passages

This research was supported by the Chosun University Research Grant, 2001

are used for determining co-occurrence

2. Information Retrieval of DNF

The Boolean model is a simple retrieval model based on set theory and Boolean algebra. Since the concept of a set is quite intuitive, the Boolean model provides a framework which is easy to grasp by a common user of an IR system. Furthermore, the queries are specified as Boolean expressions which have precise semantics.

Unfortunately, the Boolean model suffers from major drawbacks. First, its retrieval strategy is based on a binary decision criterion without any notion of a grading scale, which prevents good retrieval performance. Thus, the Boolean model is in reality much more a data retrieval model. Second, while Boolean expressions have precise semantics, frequently it is not simple to translate an information need into a Boolean expression. In fact, most users find it difficult and awkward to express their query requests in terms of Boolean expressions[3].

The Boolean model considers that index terms are present or absent in a document. As a result, the index term weights are assumed to be all binary, wi,j∈ {0,1}. A query q is composed of index terms linked by three connectives: not, and, or. Thus, a query is essentially a conventional Boolean expression which can be represented as a disjunction of conjunctive vectors. For instance, the query [q = Ka∧(Kb ∨ ¬Kc)] can be written in disjunctive normal form as [$\vec{q}$dnf =(1,1,1)∨(1,1,0)∨(1,0,0)], where each of the components is a binary weighted vector associated with the tuple(ka, kb, kc). These binary weighted vectors are called the conjunctive components of $\vec{q}$dnf.

The Boolean model predicts that each document is either relevant or non relevant. There is no notion of a partial match to the query conditions. For instance, let dj be a document for which $\vec{d}_j=(0,1,0)$. document dj includes the index term kb but is considered non-relevant to the query $[q=k_a\wedge(k_b\vee\neg k_c)]$.

Search process after convert Boolean document to disjunctive normal form and it is ranked retrieval result based on vector retrieval.

2.1 term-weight ranking

The vector model proposes to evaluate the degree of similarity of the document dj with regard to the query q as the correlation between the vectors $\vec{d}_j$ and $\vec{q}$. This correlation can be quantified, for instance, by the cosine of the angle between these two vectors.

In the vector model, intra-clustering similarity is quantified by measuring the raw frequency of a term ki inside a document dj. Such term frequency is usually referred to as the tf factor and provides one measure of how well that term describes the document contents. Furthermore, inter-cluster dissimilarity is quantified by measuring the inverse of the frequency of a term ki among the documents in the collection. This factor is usually referred to as the inverse document frequency or the idf factor. The motivation for usage of an idf factor is that terms which appear in many documents are not very useful for distinguishing a relevant document from a non-relevant one.

$$sim(d_j, q)=\frac{\vec{d}_j\cdot\vec{q}}{|\vec{d}_j|\times|\vec{q}|}=\frac{\sum_{i=1}^{t} w_{i,j}\times w_{i,q}}{\sqrt{\sum_{i=1}^{t} w_{i,j}^{2}}\times\sqrt{\sum_{i=1}^{t} w_{i,q}^{2}}} \quad (1)$$

Let N be the total number of documents in the system and ni be the number of documents in which the index term ki appears Let freqi,j be the raw frequency of term ki in the document dj. Then, the normalized frequency fi,j of term ki in document dj is qiven by

$$f_{i,j}=\frac{freq_{i,j}}{\max_l freq_{l,j}} \quad (2)$$

where the maximum is computed over all terms which are mentioned in the text of the document dj. IF the term ki does not appear in the document dj then fi,j =0. Further, let idfi, inverse document frequency for ki, be given by

$$idf_i = \log\frac{N}{n_i} \quad (3)$$

The best known term-weighting schemes use weights which are given by

$$w_{i,j} = f_{i,j}\times\log\frac{N}{n_i} \quad (4)$$

For the query term weights, Salton and Buckey suggest

$$w_{i,q} = \left(0.5+\frac{0.5\ freq_{i,q}}{\max_l freq_{l,q}}\right)\times\log\frac{N}{n_i} \quad (5)$$

where freqi,q is raw frequency of the term ki in the text of the information request q.

3. Information Retrieval model of RF and LCAF

3.1 Information Retrieval Model of Relevance Feedback

The main idea consists of selecting important terms, or expressions, attached to the documents that have been identified as relevant by the user, and of enhancing the importance of these terms in a new query formulation[4]. The expected effect is that the new query will be moved towards the relevant documents and away from the non-relevant ones.

The application of relevance feedback to the vector model considers that the term weight vectors of the

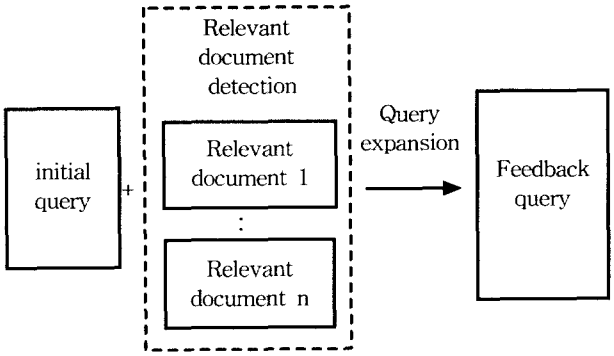

Fig. 1. General relevance feedback

documents identified as relevant have similarities among themselves. Further, it is assumed that non-relevant documents have term-weight vectors which are dissimilar from the ones for the relevant documents. The basic idea is to reformulate the query such that it gets closer to the term-weight vector space of the relevant documents.

Consider first the unrealistic situation in which the complete set Cr of relevant documents to a given query q is known in advance. In such a situation, it can be demonstrated that the best query vector for distinguishing the relevant documents from the non-relevant documents is given by

$$\vec{q}_{opt} = \frac{1}{|C_r|} \sum_{\forall \vec{d}_j \in C_r} \vec{d}_j - \frac{1}{N-|C_r|} \sum_{\forall \vec{d}_j \notin C_r} \vec{d}_j \quad (6)$$

Dr : set of relevant documents, as identified by the user, among the retrieved documents

Dn : set of non-relevant documents among the retrieved documents

Cr : set of relevant documents among all documents in the collection

|Dr|, |Dn|, |Cr| : number of documents in the sets Dr, Dn, and Cr, respectively.

α, β, γ : tuning constants.

The problem with this formulation is that the relevant documents which compose the set Cr are not known a priori. In fact, we are looking for them. The natural way to avoid this problem is to formulate an initial query and to incrementally change the initial query vector. This incremental change is accomplished by restricting the computation to the documents known to be relevant at that point. There are three classic and similar ways to calculate the modified query $\vec{q}_m$ as follows

Standard_Rocchio : (7)

$$\vec{q}_m = \alpha \vec{q} + \frac{\beta}{|D_r|} \sum_{\forall \vec{d}_j \in D_r} \vec{d}_j - \frac{\gamma}{|D_n|} \sum_{\forall \vec{d}_j \in D_n} \vec{d}_j$$

Ide_Regular : (8)

$$\vec{q}_m = \alpha \vec{q} + \beta \sum_{\forall \vec{d}_j \in D_r} \vec{d}_j - \gamma \sum_{\forall \vec{d}_j \in D_n} \vec{d}_j$$

Ide_Dec_hi : (9)

$$\vec{q}_m = \alpha \vec{q} + \beta \sum_{\forall \vec{d}_j \in D_r} \vec{d}_j - \gamma \text{maxnon\_relevant}(\vec{d}_j)$$

where maxnon_relevant ($\vec{d}_j$) is a reference to the highest ranked non-relevant document. Notice that now Dr and Dn stand for the sets of relevant and non-relevant documents according to the user judgement, respectively. In the original formulations, Rochio fixed α =1 and Ide fixed $\alpha = \beta = \gamma = 1$. The expressions above are modern variants. The current understanding is that the three techniques yield similar results

3.2 Information Retrieval Model of Local Context Analysis Feedback

In a user relevance feedback cycle, the user examines the top ranked documents and separates them into two classes, the relevant ones and the non-relevant ones. This information is then used to select new terms for query expansion. The reasoning is that the expanded query will retrieve more relevant documents. Thus, there is an underlying notion of clustering supporting the feedback strategy. According to this notion, known relevant documents contain terms which can be used to describe a larger cluster of relevant documents. In this case, the description of this larger cluster of relevant documents is built interactively with assistance from the user.

In a local strategy, the documents retrieved for a given query q are examined at query time to determine terms for query expansion[5]. This is similar to a relevance feedback cycle but might be done without assistance from the user.

Distinct automatic approaches for selecting index terms can be used. A good approach is the identification of noun groups which we now discuss.

A sentence in natural language text is usually composed of nouns, pronouns, articles, verbs, adjectives, adverbs, and connectives. While the words in each grammatical class are used with a particular purpose, it can be argued that most of the semantics is carried by the noun words. Thus, an intuitively promising strategy for selecting index terms automatically is to use the nouns in the text. This can be done through the systematic elimination of verbs, adjectives, adverbs, connectives, articles, and pronouns.

Since it is common to combine two or three nouns in a single component, it makes sense to cluster nouns which appear nearby in the text into a single indexing component. Thus, instead of simply using nouns as index terms, we adopt noun groups. A noun group is a set of nouns whose syntactic distance in the text(measure in terms of number of words between two nouns) does not exceed a predefined threshold(for instance, 3).

When noun groups are adopted as indexing terms, we obtain a conceptual logical view of the documents in terms of sets of non-elementary index terms.

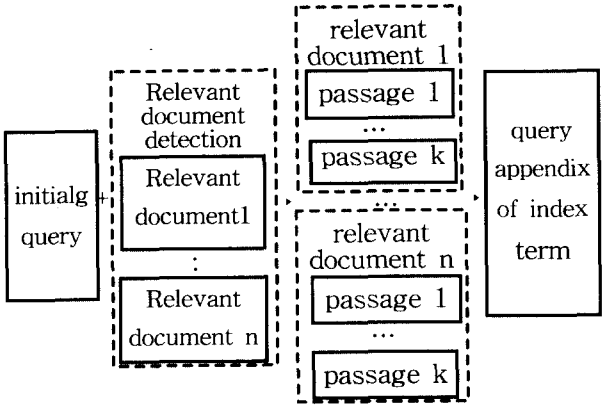

Fig. 2. Information retrieval model using local context analysis

More specifically, the local context analysis procedure operates in three steps.

- First, retrieve the top n ranked passages using theoriginal query. This is accomplished by breaking up the documents initially retrieved by the query in fixed length passages and ranking these passages as if they were documents.
- Second, for each concept c in the top ranked passages, the similarity sim(q, c) between the whole query q and the concept c is computed using a variant of tf-idf ranking.
- Third, the top m ranked concepts are added to the original query q. To each added concept is assigned a weight given by 1-0.9 ×i/m where i is the position of the concept in the final concept ranking. The terms in the original query q might be stressed by assigning a weight equal to 2 to each of them.

The second on is the most complex and the one which we now discuss.

The similarity sim(q, c) between each related concept c and the original query q is computed as follows.

$$sim(q,\ c) = \prod_{k_i \in q}\left(\delta + \frac{\log\left(f(c, k_i) \times idf_c\right)}{\log n}\right)^{idf_i} \tag{10}$$

where n is the number of top ranked passages considered. The function f(c, ki) quantifies the correlation between the concept c and the query term ki and is given by

$$f(c, k_i) = \sum_{j=1}^{n} pf_{i,j} \times pf_{c,j} \tag{11}$$

where $pf_{i,j}$ is the frequency of term ki in the j-th passage and $pf_{c,j}$ is the frequency of the concept c in the j-th passage. Notice that this is the standard correlation measure defined for association clusters but adapted for passages. The inverse document frequency factors are computed as

$$idf_i = \max\left(1, \frac{\log_{10} N/npi}{5}\right) \tag{12}$$

$$idf_c = \max\left(1, \frac{\log_{10} N/npc}{5}\right) \tag{13}$$

where N is the number of pasages in the collection, npi is the number of passages containing the term ki, and npc is the number of passages containing the concept c.

The factor δ is a constant parameter which avoids a value equal to zero for sim(q, c). Usually, δ is a small factor with values close to 0.1. Finally, the idfi factor in the exponent is introduced to emphasize infrequent query terms.

4. Experimentation and Result

When considering retrieval performance evaluation, we should first consider the retrieval task that is to be evaluated.

Consider and example information request I and its set relevant documents Let |R| be the number of documents in this set. Assume that a given retrieval strategy processes the information request I and generates a document answer set A. Let |A| be the number of documents in this set. Further, let |Ra| be the number of documents in the intersection of the sets R and A.

- Recall is the fraction of the relevant documents (the set R) which has been retrieved

$$\text{Recall} = \frac{|Ra|}{|R|} \tag{14}$$

- Precision is the fraction of the retrieved documents (the set A) which is relevant

$$\text{Precision} = \frac{|Ra|}{|A|} \tag{15}$$

A single measure which combines recall and precision might be of interest. One such measure is the harmonic mean F of recall and precision which is computed as

$$F(j) = \frac{2}{\frac{1}{r(j)} + \frac{1}{P(j)}} \tag{16}$$

where r(j) is the recall for the j-th document in the ranking, P(j) is the precision for the j-th document in the ranking, and F(j) is the harmonic mean of r(j) and P(j). The function F assumes values in the interval [0, 1]. It is 0 when no relevant documents have been retrieved and is 1 when all ranked documents are relevant. Further, the harmonic mean F assumes a high value only when both recall and precision are high. Therefore, determination of the maximum value for F can be interpreted as an attempt to find the best possible compromise between recall and precision.

There is experimentation result comparison for DNF initial and relevance feedback retrieval in Table 1, it is show that relevance feedback retrieval result more improvement 61.45% at recall, 58.7% at precision for

initial retrieval result.

Table 1. Experimentation result comparison for DNF initial and relevance feedback retrieval

division / measure	DNF initial	Relevance feedback	Increase rate
Recall	0.39	0.63	+0.24(+61.54)
precision	0.46	0.73	+0.27(+58.70)

Table 2. DNF initial and relevance feedback recall (document number limit)

division / measure	Recall		
	DNF initial	Relevance feedback	Increase rate
document number≤10	0.28	0.48	+0.20(+71.43)
document number≤20	0.51	0.78	+0.27(+52.94)

Table 3. DNF initial and relevance feedback precision (document number limit)

division / measure	Precision		
	DNF initial	Relevance feedback	Increase rate
document number≤10	0.47	0.75	+0.28(+59.58)
document number≤20	0.44	0.71	+0.27(+61.36)

Fig. 3. show that retrieval experimentation result for DNF initial and relevance feedback when retrieval document limict 20. This Fig. 3. also show that precision wasn't change largely at 0.6 point at recall for all result of DNF initial and relevance feedback retrieval

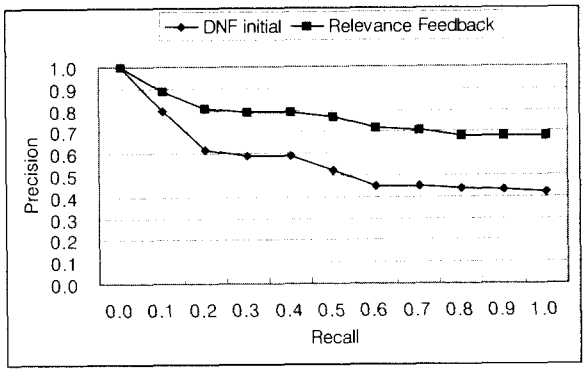

Fig. 3. Retrieval experimentation result for DNF initial and relevance feedback

A single measure which combines recall and precision might be of interest.

Table 6. show that result comparison of relevance feedback(RF) and local context analysis feedback(LCAF) retrieval experimentation, LCAF result is more improve of 3.173% recall, 12.82% precision for relevance feedback retrieval result.

Table 4. Single Measure appraisement for DNF initial retrieval using harmonic mean

Recall	Precision	Harmonic Mean	Total Harmonic Mean
0.1	0.8	0.18	0.4636
0.2	0.62	0.32	
0.3	0.59	0.39	
0.4	0.59	0.465	
0.5	0.52	0.476	
0.6	0.45	0.514	
0.7	0.45	0.548	
0.8	0.44	0.568	
0.9	0.43	0.584	
1.0	0.42	0.591	

Table 5. Single Measure appraisement for relevance feedback retrieval using harmonic mean

Recall	Precision	Harmonic Mean	Total Harmonic Mean
0.1	0.89	0.18	0.5758
0.2	0.81	0.327	
0.3	0.79	0.435	
0.4	0.79	0.531	
0.5	0.77	0.606	
0.6	0.72	0.653	
0.7	0.71	0.704	
0.8	0.68	0.735	
0.9	0.68	0.778	
1.0	0.68	0.809	

Table 6. Result comparison of RF and LCAF retrieval experimentation

division / measure	RF	LCAF	Increase rate
Recall	0.63	0.65	+0.02(+3.174)
Precision	0.73	0.78	0.05(+12.82)

Table 7. Recall of RF and LCAF (document number limit)

division / measure	Recall		
	RF	LCAF	Increase rate
document number≤10	0.48	0.52	+0.04(+8.33)
document number≤20	0.78	0.79	+0.01(+1.282)

Table 8. Precision of RF and LCAF (document number limit)

division / measure	Precision		
	RF	LCAF	Increase rate
document number≤10	0.75	0.77	+0.02(+2.67)
document number≤20	0.71	0.74	+0.03(+4.23)

Fig. 4. show that experimentation of RF and LCAF retrieval when retrieval document limit 20.

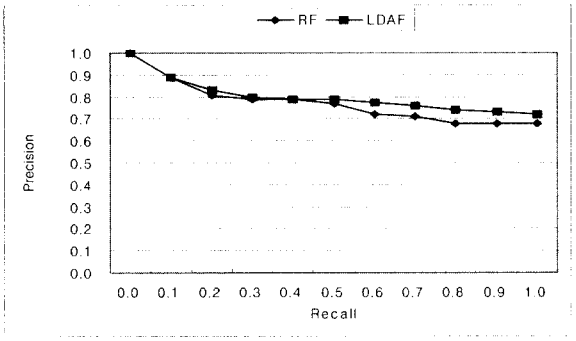

Fig. 4. Experimentation result of RF and LCAF retrieval

A single measure which combines recall and precision might be of interest.

Table 9. Single measure appraisement of RF using harmonic mean

Recall	Precision	Harmonic Mean	Total Harmonic Mean
0.1	0.89	0.18	0.5758
0.2	0.81	0.327	
0.3	0.79	0.435	
0.4	0.79	0.531	
0.5	0.77	0.606	
0.6	0.72	0.653	
0.7	0.71	0.704	
0.8	0.68	0.735	
0.9	0.68	0.778	
1.0	0.68	0.809	

Table 10. Single measure appraisement of LCAF using harmonic mean

Recall	Precision	Harmonic Mean	Total Harmonic Mean
0.1	0.89	0.180	0.5898
0.2	0.831	0.322	
0.3	0.8	0.437	
0.4	0.79	0.531	
0.5	0.79	0.612	
0.6	0.773	0.675	
0.7	0.76	0.728	
0.8	0.74	0.769	
0.9	0.729	0.809	
1.0	0.72	0.837	

5. Conclusion

Local analysis techniques are interesting because they take advantage of the local context provided with the query. In this regard, they seem more appropriate than global analysis techniques. Furthermore, many positive results have been reported in the literature. The application of local analysis techniques to the Web, however, has not been explored and is a promising research direction.

References

[1] A. Bookstein. Implication of Boolean structure for probabilistic retrieval. In Proc. of the 8th Annual International ACM/SIGIR Conference on Research and Development in Information Retrieval, pages 11-17, Montreal, Canada, 1995.

[2] E. A. Fox. Extending the Boolean and Vector space Models of Information Retrieval with P-Norm Queries and Multiple Concept Types. PhD thesis, Cornell University, Ithaca, New York, Http:// www.ncstrl.org, 1983.

[3] Donna Harman. Relevance feedback revisited. In Proc. of the 5th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 1-10, Copenhagen, Denmark, 1992.

[4] J. Xu and W. B. Croft. Query expansion using local and global document analysis. In Proc. ACM-SIGIR Conference on Research and Development in Information Retrieval, pages 4-11, Zurich, Switzerland, 1996.

[5] Baeza-Yates, R. and Ribeiro-Neto, Berthier. Moderm Information Retrieval, addison-wesley Pub. Co(sd), 1992.

[6] Mandar Mitra, Amit Singhal, and Chris Buckley. Improving automatic query expansion. In B. Croft, A. Moffat, C. J. van Rijsbergen, R. Wilkinson, and J. Zobel, editors, Proc. of 21st Annual Int. ACM SIGIR Conference on Research and Development in Information Retrieval, pages 206-214, Melbourne, Australia, 1998..

Sung-Joo Lee

1970 : Dept. of Physics Sciences, Han nam University(B.S)

1992 : Dept. of Computer sciences, Kwangwoon University(M.S)

1998년 : Dept. of Computer sciences, Catholic University of Daegu(Ph.D)

1988~1990 : Chief, Computer Center, Chosun University

1995~1997 : President, Information Science, Chosun University

1981~now : Professor in the Dept. of Computer Engineering, Chosun University

Research Interests : Software engineering, Programming Language, Object-oriented software, Rough set.

You-Mi Moon

1983 : Dept. of Computer Science, Chosun University(B.S)
1987 : Dept. of Computer Science, Chosun University(M.S)
1998 ~ now : Dept. of Computer Science, Chosun University Doctoral Student

Research Interests : Software engineering(Reuse, metrics) Object-oriented software(metrics) Rough set, Fuzzy set Electronic Commerce
e-mail : mym60@hananet.net

Young-Chon Kim

1992 : Dept. of Computer Science, Kwangju University(B.S)
1996 : Dept. of Computer Engineering, Chosun University(M.S)
1998~now : Dept. of Computer Science, Chosun University Doctoral Student

Research Interests : Software engineering(Reuse, metrics), Object-oriented software(metrics), Electronic Commerce, InformationRetrieval.