# 엔트리 페이지 검색을 위한 메타 검색

강 인 호†


## 요 약

본 연구에서는 웹에서 사용자가 방문하고자 하는 곳을 찾아가는 엔트리 페이지 검색을 위한 메타검색 방식을 제안한다. 기존의 연구에서 메타 검색이 여러 검색 엔진에서 많이 나타나는 중복된 문서를 강조하는 방식인 반면에 비해, 본 연구에서는 문서의 중복 개념을 확장하여 특정 도메인 및 디렉토리에서 나온 문서들도 중복되었다고 가정하여 메타검색에 이용하는 방식을 보인다. TREC에 제출된 시스템들의 결과물과 상용 검색 엔진의 결과물을 이용하여, 확장된 중복을 이용한 메타 검색의 유용성을 실험한다. 수행된 실험을 통해서 문서의 단순 중복을 이용하는 기존의 방식이 내용 기반 검색에 유용한 반면, 엔트리 페이지 검색에 있어서는 본 연구에서 제안하는 확장된 중복 방식이 기존 방식의 성능보다 10% 이상의 성능 향상을 얻을 수 있음을 알 수 있었다.


# MetaSearch for Entry Page Finding Task

In-Ho Kang†


## ABSTRACT

In this paper, a MetaSearch algorithm for navigational queries is presented. Previous MetaSearch algorithms focused on informational queries. They gave a high score to an overlapped document. However, the overemphasis of overlapped documents may degrade the performance of a MetaSearch algorithm for a navigational query. However, if a lot of result documents are from a certain domain or a directory, then we can assume the importance of the domain or directory. Various experiments are conducted to show the effectiveness of overlap of a domain and directory names. System results from TREC and commercial search engines are used for experiments. From the results of experiments, the overlap of documents showed the better performance for informational queries. However, the overlap of domain names and directory names showed the 10% higher performance for navigational queries.


**키워드 : MetaSearch(메타 검색), 내용 기반 검색(Informational Query), 엔트리 페이지 검색(Navigational Query), 도메인명 중복(Domain Name Overlap)**

## 1. 서 론

검색 엔진은 사용자가 원하는 정보를 효과적으로 찾기 위한 웹 검색 기법을 고안하여 제작한다. 하지만 하나의 검색 엔진이 모든 질의에 대해서 좋은 성능을 보이는 것은 아니다. 특정 상황에서 좋은 성능을 보이는 검색 기법이라 하더라도 다른 상황에서는 그렇지 못한 경우도 있다[1][2]. 예를 들어, 웹 문서 내용 위주의 검색은 질의로 사용되는 입력 단어의 다양함과 웹 문서의 질에 영향을 받으며, 페이지 랭크와 같은 링크 정보는 불완전한 혹은 의미 없는 링크에 의해 성능 감소를 초래한다. 각 검색 기법의 장점을 살리고 단점을 보완할 수 있도록 여러 검색 엔진의 결과를 결합하는 방법이 연구 되는데, 이러한 검색 방법을 메타 검색(MetaSearch)이라고 한다[3][4][5].

사용자가 자신이 요구하는 정보에 대하여 기술한 질의를 웹 검색 엔진에 입력하면, 각 검색 엔진은 그 질의와 관련 있는 문서를 순위화된 리스트의 형태로 출력하여 보여준다. 메타 검색 엔진은 각 검색 엔진이 산출한 순위화 된 리스트들을 입력으로 받아서 메타 검색 알고리즘을 적용한 후 하나의 순위화된 리스트를 결과로 낸다. 어떤 순위화 알고리즘을 사용하여 여러 문서 리스트들을 결합하느냐에 따라 메타 검색 엔진의 성능이 좌우된다[3][6][7][8][9] .

웹 환경에서의 사용자 질의는 검색 대상과 목적이 무엇이냐에 따라 다음과 같이 구분이 가능하다[10].

- 내용 기반 검색 (informational query)
- 엔트리 페이지 검색 (navigational query)

내용 기반 검색의 경우, 사용자가 알고자 하는 정보를 설명하는 혹은 정보와 관련된 문서를 찾는다. 이러한 경우의 정답은 웹 문서 자체가 되며, 여러 개의 정답이 존재할 수 있다. 반면 엔트리 페이지 검색은 사용자가 관심 있어 하는 개인이나 단체의 사이트 입구를 찾는다. 일반적으로 웹 사

† 정 회 원 : 삼성종합기술원 Computing LAB 전문연구원
논문접수 : 2004년 9월 14일, 심사완료 : 2005년 3월 15일

이트에는 주된 입구가 되는 엔트리 페이지(entry page)가 존재한다. 엔트리 페이지에는 그 웹 사이트에 대한 간단한 소개와 사이트를 구성하고 있는 다른 웹 문서로의 링크가 있다. 엔트리 페이지는 메인 페이지 혹은 홈페이지라고도 일컬어 진다. 예를 들어 "한국 과학기술원 산업디자인과"라는 질의를 입력하는 사용자는 그 사이트의 엔트리 페이지인 *http://id.kaist.ac.kr/index.htm*으로 가기를 원한다. 하지만, 동일 호스트 (*http://id.kaist.ac.kr*)에도 여러 사이트의 웹 문서들이 있을 수 있다. "한국과학기술원 산업디자인과 제품환경체계 연구실"이라는 질의에 대해서는 *http://id.kaist.ac.kr/pne/main.htm*이라는 엔트리 페이지가 정답이 된다. 즉 사이트의 엔트리 페이지는 호스트의 최상위 디렉터리의 URL을 뜻하는 것만은 아니다.

이와 같이 검색 대상과 목적이 다양해지고 있지만, 기존의 메타 검색에 대한 연구는 내용 기반 검색의 결과를 어떻게 잘 결합할 것인가에 초점이 맞추어져 있다. 사용자가 어떤 사이트의 엔트리 페이지를 방문하고자 하는 경우의 메타 검색에 관해서는 연구가 이루어지지 않고 있다. 하나의 검색 엔진은 모든 상황에서 좋은 성능을 보이지는 않으며 제한된 범위를 가지므로, 내용기반 검색에서와 마찬가지로 엔트리 페이지 검색에서도 여러 검색 엔진의 결과를 결합하는 것이 필요하다. 본 연구에서는 문서의 중복 개념을 확장하여 특정 도메인에서 나온 문서들도 중복되었다고 가정하여 사이트 검색에 적용하는 방법을 보인다.

## 2. 관련연구

본 절에서는 메타 검색의 개념을 정의하고 그 종류를 살펴본다.

### 2.1 메타 검색

우선 $n$개의 검색엔진이 출력하는 결과물에 대해서 다음과 같이 정리한다. $r_i(d)$는 $i$번째 검색 엔진이 출력한 문서 $d$의 순위를 나타낸다. 그리고 $s_i(d)$는 $i$번째 검색 엔진이 출력한 문서 $d$의 질의와의 유사도(매칭 정도)를 나타낸다. $d$와 $i$가 명확한 경우에는 생략해서 표현한다. 예를 들면 $r(d)$, $s(d)$, $r_i$, $s_i$와 같이 사용한다. 예를 들어 3개의 검색 엔진($S_1$, $S_2$, $S_3$)이 다음과 같이 결과를 제시했다고 하자[9].

- $S_1$: ($d_1$, -5.05937), ($d_{43}$, -5.19892), ($d_{162}$, -5.25427), …
- $S_2$: ($d_{43}$, -3.93243), ($d_{14}$, -4.11519), ($d_{501}$, -4.30411), …
- $S_3$: ($d_{67}$, 18.420), ($d_{923}$, 18.292), ($d_{501}$, 18.051), …

$S_1$ 검색 엔진은 $d_1$을 1순위로 제출했고, $d_{43}$을 2순위로 그리고 $d_{162}$를 3순위로 제시하였다. 또한 $d_1$에 대해서 사용자 질의와의 유사도의 값으로 5.05937를 출력했고, $d_{43}$에 대해서는 5.19892를 출력했다. 이를 표현하면 $r_1(d_1)=1$ $s_1(d_{43})=-5.19892$ 등과 같이 작성할 수 있다. $i$번째 검색 엔진이 제시한 결과 문서 집합은 $D_i$로 표현하며 $D$는 메타 검색에 참여한 모든 검색 엔진의 결과 문서 집합을 합친 결과이다 ($D=\cup D_i$). $D$에 속한 문서 중 $D_i$에 속하지 않는 문서들에 대해서 $i$번째 검색 엔진은 $D_i$의 그 어떤 문서의 유사도보다도 낮은 값을 할당했다고 가정한다. 여러 검색 엔진에서 제공하는 유사도를 사용하기 위해서 [0, 1] 사이의 값으로 정규화해서 사용한다. 문서 $d$의 유사도를 정규화하는 일반적인 공식은 수식 1과 같다[6][7].

$$s'(d)=\frac{s(d)-\min(s)}{\max(s)-\min(s)} \qquad \text{(수식 1)}$$

여기에서 $min(s)$와 $max(s)$는 유사도로 가지는 값 중 제일 작은 값과 제일 큰 값을 의미한다. 이렇게 선형적인 형태의 유사도 정규화 외에도 학습을 통한 유사도의 분포 형태를 추정하는 연구도 있다 [8][9][11]. 본 연구에서는 학습 데이터 없이 비지도 학습을 기반으로 메타 검색을 수행한다.

### 2.2 결합 함수

[3]은 각 검색 엔진이 제공하는 유사도의 최소값, 최대값, 중앙값, 평균값, 합을 이용하여 각 문서를 재순위하였다. 그들은 또한 검색 엔진에서 나타난 횟수에 따라서 가중치를 두는 방식 또한 연구하였다.

〈표 1〉 결합 함수의 예

| 결합 함수 | 결합 후의 유사도 |
|---|---|
| *CombMIN* | 각 검색 엔진이 산출한 유사도 중 최소값 |
| *CombMAX* | 각 검색 엔진이 산출한 유사도 중 최대값 |
| *CombMED* | 각 검색 엔진이 산출한 유사도 중 중앙값 |
| *CombSUM* | 각 검색 엔진이 산출한 유사도의 총합 |
| *CombANZ* | CombSUM ÷ (해당 문서를 결과로 낸 검색엔진의 개수) |
| *CombMNZ* | CombMNZ × (해당 문서를 결과로 낸 검색엔진의 개수) |

메타 검색을 위해 사용한 결합 함수는 수식 2와 같이 나타낼 수 있다[9].

$$s(d)=n(d)^{\gamma}\sum_i s_i(d) \qquad \text{(수식 2)}$$

여기에서 $n(d)$는 문서 $d$를 결과로 제시한 검색 엔진의 개수를 나타내며, $\gamma$는 $\{-1, 0, 1\}$의 값을 가진다. 예를 들어 $\gamma$가 $1$의 값을 가지면 *CombANZ*, $\gamma$가 $0$의 값을 가지면 *CombSUM* 그리고 $\gamma$가 $1$의 값을 가지면 *CombMNZ*를 나타낸다. [6]은 두 검색 엔진의 결과를 바탕으로 메타 검색을 할 때 정답 문서와 비정답 문서가 두 검색 결과에 공통적으로 나타나는 비율을 계산하였다. 그는 '서로 다른 검색 엔진은 서로 다른 집합의 정답이 아닌 문서를 검색하지만, 유사한 집합의 정답 문서를 검색한다'는 것을 발견하였다. *CombMNZ*는 여러 검색 엔진이 결과로 제시한 문서에 가

중치를 두기 위해서 *CombSUM*값을 조정한 것으로, 그가 주장하는 이론과 부합하는 결합 함수이다. 그는 $\gamma$의 값으로 *{0, 0.5, 1.2, 5, 10}*을 사용하여 실험한 결과, $\gamma$가 *1*인 *CombMNZ*가 가장 좋은 성능을 보임을 보였다. 이로써, 여러 검색 엔진에 공통으로 나온 문서가 높은 순위를 얻을 수 있도록 하는 *CombMNZ*는 메타 검색에 간단하면서도 가장 효과적인 방법으로 자리잡았다.

### 2.3 TREC - 테스트 컬렉션

TREC(Text REtrieval Conference)은 NIST에 의해 매년 개최되는 문서 검색에 관한 학술 대회이다. 그 중 Web Track은 웹 상에서의 정보 검색 기량을 겨루는 대회이다. 이전까지의 TREC Web Track에서는 내용 기반 검색에 해당하는 adhoc task만 있었으나, 2001년부터는 사용자 요구의 다양성을 고려하여 엔트리 페이지 검색에 해당하는 entry page finding task를 추가하였다[12]. 각 검색 엔진은 한 질의에 대해서 adhoc task의 경우에는 최대 1,000개, entry page finding task의 경우에는 최대 100개까지 결과를 내고 있다. 본 연구에서는 TREC-2001 Web Track의 adhoc task와 entry page finding task에 제출된 시스템들의 결과를 대상으로 메타 검색을 수행한다.

## 3. 엔트리 페이지 검색을 위한 메타 검색

본 절에서는 내용 기반 검색과 엔트리 페이지 검색의 차이점을 살펴보고 엔트리 페이지 검색을 위한 결합 함수를 제안한다.

### 3.1 중복의 단위

<표 2>는 TREC-2001 Web Track에 제출된 내용 정보 검색 엔진과 홈페이지 검색 엔진에 대해서 질의당 정답 문서의 개수를 비교한 것이다.

〈표 2〉 TREC-2001 Web Track의 구성

| | 참가 엔진 개수 | 질의 개수 | 질의당 정답 문서 개수 |
|---|---|---|---|
| 내용 기반 검색 | 97 | 501-550 (50개) | 67.26 |
| 엔트리 페이지 검색 | 43 | 1-145 (145개) | 1.74 |

내용 기반 검색의 경우 질의당 정답 문서의 개수는 67.26개, 엔트리 페이지 검색의 경우는 1.74개이다. 분석 결과, 엔트리 페이지 검색의 정답이 두 개 이상인 경우는 어떤 사이트의 미러 사이트(mirror site)이거나 URL 재지정(redirection)에 의한 중복 페이지로 인한 것이다. 이러한 경우를 제외하면 엔트리 페이지 검색에서 질의에 적합한 문서의 개수는 대부분 한 개이다. 이와 같이 엔트리 페이지 검색은 내용 검색의 경우보다 정답의 개수가 훨씬 적다. 엔트리 페이지 검색에 *CombMNZ*를 적용한다고 할 경우, 이는 중복되는 비정답 문서에 더 가중치를 줄 가능성이 있다.

검색 엔진에게 있어 정확한 엔트리 페이지를 찾는 것은 힘이 들 수 있다. 그러나 엔트리 페이지와 관련된 혹은 비슷한 웹 문서를 찾는 것은 힘이 덜 든다. 엔트리 페이지 검색에 있어서 특정 도메인의 웹 문서들이 여러 검색 엔진에서 나타날 경우, 우리는 정답 엔트리 페이지가 그 도메인에 속할 가능성이 높다는 것을 유추할 수 있다.

<표 3>은 두 검색 엔진에 중복하여 나타난 문서 혹은 도메인이 정답일 비율을 나타낸다.

〈표 3〉 중복 단위에 따른 정확률의 비교

| Overlap Unit | | 내용 기반 검색 | 사이트 검색 |
|---|---|---|---|
| document | $\frac{\lvert R \cap D_1 \cap D_2 \rvert}{\lvert D_1 \cap D_2 \rvert}$ | 0.11 | 0.10 |
| domain name | $\frac{\lvert RD \cap DD_1 \cap DD_2 \rvert}{\lvert DD_1 \cap DD_2 \rvert}$ | 0.14 | 0.33 |

R: 정답 문서 집합
Di: i번째 결과 문서
RD: 정답 도메인 집합
DDi: i번째 결과 문서의 도메인

TREC-2001에 참여한 검색 엔진 중 임의의 두 엔진을 선정하여, 두 검색 엔진에서 공통적으로 나타나는 것이 정답일 비율을 조사해 표 3에 나타내었다. 임의의 두 검색 시스템을 선정하는 작업은 100번 수행되었으며, 표 3에 나타난 값은 그 100번의 결과를 평균한 것이다. $\lvert R \cap D_1 \cap D_2 \rvert / \lvert D_1 \cap D_2 \rvert$는 문서를 단위로 두 검색 엔진의 결과를 살펴, 두 검색 엔진에 공통으로 나타난 문서가 정답인 비율을 조사한 것이고, $\lvert RD \cap DD_1 \cap DD_2 \rvert / \lvert DD_1 \cap DD_2 \rvert$는 단위를 도메인 단위로 확장하여 두 검색 엔진의 결과를 살펴, 두 검색 엔진에서 공통으로 나타난 도메인명이 정답인 비율을 조사한 것이다. 여기서 도메인 단위로 확장이란, *http://www.radio.cbc.ca:80/radio/programs/current/quirks/index.html*와 같이 *http://www.radio.cbc.ca/*라는 도메인 명을 포함하고 있는 URL의 문서는 모두 *http://www.radio.cbc.ca/*라는 도메인에 속한다고 보는 것을 뜻한다. 물론 사이트는 도메인 이름대로 결정되는 것은 아니다. 하지만 본 연구에서는 같은 도메인 명을 가지는 문서는 모두 같은 사이트에 속한다고 가정한다. 표 3에서, 두 내용 검색 엔진에 공통적으로 나타나는 문서 중 정답인 문서의 비율과 두 엔트리 페이지 검색 엔진에 공통적으로 나타나는 문서 중 정답인 문서의 비율은 큰 차이가 없다. 그러나 문서 단위에서 도메인 명 단위로 확장하여 그 비율을 비교하면 각각 0.14, 0.33으로 차이가 크다. 이러한 특성은 엔트리 페이지 검색의 결과를 결합할 때에는 문서 단위로 중복된 것뿐만 아니라, 도메인 단위로 중복의 정도를 파악하는 것도 의미 있는 일임을 알 수 있다.

### 3.2 SiteSUM

본 연구에서 제안하는 확장된 중복 단위를 고려하기 위해 결과 문서의 URL을 분해한다. URL은 도메인명과 디렉터리 단위로 분해된다. 확장된 중복 단위의 점수는 각 디렉터리 별로 계산이 된다. 이는 디렉터리 별로 많은 문서를 포함하는 디렉터리를 강조하기 위함이다. 즉 상위 디렉터리의 문서는 하위 디렉터리 문서의 유사도를 흡수한다. 그래서 동일한 깊이[1]의 디렉터리명이라도 서로 다른 유사도를 가진다. 이러한 내용을 고려한 *SiteSUM*은 수식 3과 같이 계산된다.

$$
\begin{aligned}
SiteSUM(d) &= \alpha \times score_{doc}(d) + \beta \times score_{site}(H) \\
current\_direct(d) &= H \\
score_{doc}(d) &= CombSUM(d) \\
score_{site}(H) &= \sum_{i=1}^{n} \sum_{\{x \mid \forall x, H \in direct(x)\}} s_i(x) \\
\alpha = 1, \quad \beta &= \frac{\max(score_{doc})}{2}
\end{aligned}
$$

(수식 3)

*SiteSUM*은 문서 단위의 유사도($score_{doc}$)와 도메인명 단위의 유사도($score_{site}$)의 합으로 표현된다. 두 유사도의 비율($\alpha$, $\beta$)은 $score_{doc}$의 최고값을 기준으로 정해진다. *current_direct(d)*는 웹 문서 *d*의 디렉터리명을 뜻한다. 그리고 *direct(d)*는 웹 문서 *d*의 현재 디렉터리명에서부터 도메인명까지 상위 디렉터리명을 경우 별로 다 포함한 집합을 말한다. 예를 들어 URL이 *http://www.radio.cbc.ca:80/radio/program.html*인 웹 문서는 *current_direct*의 값으로 *http://www.radio.cbc.ca/radio* 를 가진다. 그리고 direct는 *{http://www.radio.cbc.ca/, http://www.radio.cbc.ca/radio}*를 결과 집합으로 가진다. 도메인명 단위의 유사도($score_{site}$)는 문서 단위 유사도와 결합되기 전에 수식 4를 이용해 [0, 1]의 값으로 정규화된다.

$$
score'_{site}(H) = \frac{score_{site}(H) - \min(score_{site})}{\max(score_{site}) - \min(score_{site})}
$$

(수식 4)

여기에서 $max(score_{site})$와 $min(score_{site})$는 각각 도메인명 단위의 유사도의 최고값과 최저값을 의미한다. *SiteSUM*을 수행하는 과정은 먼저 기존 메타 검색과 동일한 형태로 문서 단위의 유사도를 계산한다. 그 후 문서 단위의 유사도를 이용하여 각 디렉터리별로 도메인 단위 유사도를 계산한다. 그 후 문서 단위의 유사도의 최고값에 맞추어 도메인 단위의 유사도를 결합하여 재순위화한다. 이는 도메인 단위 유사도를 이용해서 엔트리 페이지가 속한 도메인을 찾아내고, 그 도메인에서의 구체적인 위치는 URL의 깊이와 문서 단위의 중복도 그리고 검색 엔진이 제시한 순위로 찾아내는 것을 의미한다. 이는 해당 도메인의 최상위 디렉터리의 문서만을 선호하지 않기 위함이다.

1) URL에서의 '/' 개수

## 4. TREC 제출 엔진을 이용한 실험

본 절에서는 *SiteSUM*의 성능을 보이기 위해서 TREC-2001 Web Track에 제출된 시스템들의 결과물을 이용한다.

### 4.1 실험 환경

실험은 크게 다음의 두 가지 방법으로 진행한다.

- 검색 엔진 무작위 선택 실험(radon sampling experiment)
- 최고 성능 검색 엔진 선택 실험(best-to-worst sampling experiment)

다양한 성능의 검색 엔진의 결과를 결합하였을 때에도 메타 검색 알고리즘이 안정적인 결과를 내는지 알아보기 위해서 검색 엔진 무작위 선택 실험을 수행한다. 제출된 여러 시스템 중에서 $n$개 (단, $n \in \{2,3,4,5\}$)를 임의로 선택한 후, 이들을 제안한 모델로 결합하여 그 성능을 측정한다. 임의로 입력 검색 엔진을 선택하는 과정을 100번 반복 시행한 후 성능 향상치를 평균한다. 그리고 일반적으로 결합에 이용되는 검색 엔진들은 비교적 안정적이면서 좋은 성능을 내는 것들이다. 검색 능력이 뛰어난 엔진들을 대상으로 메타 검색을 할 때의 효과를 알아보기 위해서 최고 성능 검색 엔진 선택 실험을 한다. Web Track에 제출된 엔진 중에는 동일 기관에서 다수의 검색 엔진을 제출한 경우도 있다. 이 실험에서는 동일 기관에서 제출된 여러 시스템 중에서 가장 좋은 성능을 보이는 엔진만을 사용하였다. tnout10epCAU, jscbtawsep2, yehp01, UniNEep1, IBMHOMER가 최고의 성능을 보이는 상위 5개의 엔진들이다. 이 실험에서는 이들 시스템을 차례로 $n$개(단, $n \in \{2,3,4,5\}$)씩 선택하여 결합한다[12].

메타 검색의 성능은 검색 엔진의 성능 향상치를 이용한다. $n$개의 입력 엔진 중에서 가장 뛰어난 엔진의 성능을 $P_b$라고 하고, 그 $n$개의 입력 엔진을 결합한 메타 검색 엔진의 성능을 $P_f$라 하였을 때, 향상치 I는 수식 5로 계산한다[9].

$$
I = \frac{P_f - P_b}{P_b}
$$

(수식 5)

검색 엔진의 성능은 내용 기반 검색과 엔트리 페이지 검색을 위해서 각각 average precision과 MRR을 사용한다. 여기서 average precision은 검색 엔진의 결과 리스트에서 정답 문서가 나타날 때마다 그 상위 순위까지의 정확도를 측정하여 평균한 것으로 (수식 6), TREC-2001 Web Track의 내용 기반 검색을 위해서 사용되었다[12].

$$P_{avg} = \frac{1}{|R|} \times \sum_{d \in R} \frac{|R_{\le r(d)}|}{r(d)} \qquad \text{(수식 6)}$$

여기에서 $R$은 정답 문서 집합을 나타내며 $R_{\le r(d)}$는 $r(d)$보다 상위로 제출된 정답 문서 집합을 말한다. $n$개의 질의에 대해서는 이 값의 평균값을 사용한다. Average precision의 평균값을 편의상 average precision이라고 한다. 그러나 정답 집합을 모르는 경우에는 $P@m$의 결과를 사용하기도 한다. 이는 $m$개의 상위 문서 중 정답인 문서의 비율을 나타내는 것이다. 주로 $P@1$, $P@5$, $P@10$을 많이 사용한다. 이 수치를 이용할 경우, 첫 번째 결과 페이지에 정답이 몇 개나 있는지를 알 수 있다.

MRR은 검색엔진이 첫 번째 정답 문서를 몇 위로 제시했는지를 측정한다. $n$개의 질의에 대해서 MRR을 측정할 때는 수식 7과 같이 각 질의의 결과에 대해 첫 번째 정답 문서의 순위 $r$의 역수를 구한 후 이 값을 평균한다 [12].

$$MRR = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{\text{answer}_i \text{ rank}} \qquad \text{(수식 7)}$$

여기서 $answer_i$ $rank$는 $i$번째 질의의 결과에 대해 첫 번째 정답 문서의 순위를 말한다. MRR에 더해서 상위 $m$개의 결과 중 정답이 포함되지 않은 경우의 비를 보이는 *%fail*을 사용하기도 한다. 상위 10개에 대해서 고려한다면, 10개의 상위 검색 결과 안에 정답이 포함되지 않은 문제의 비를 나타낸다.

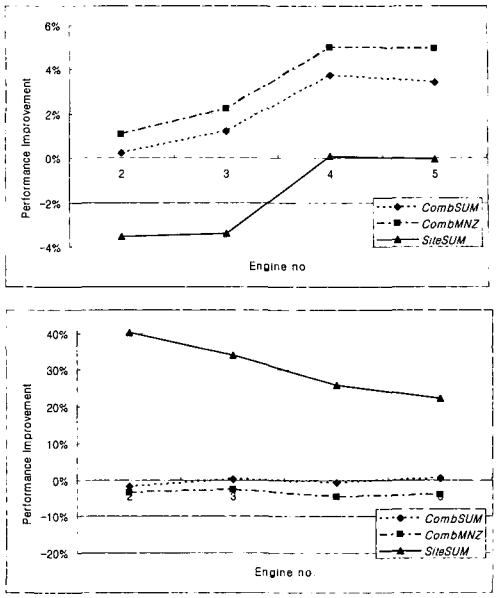
### 4.2 실험 결과

(그림 1)은 검색 엔진 무작위 선택 실험의 결과이다.

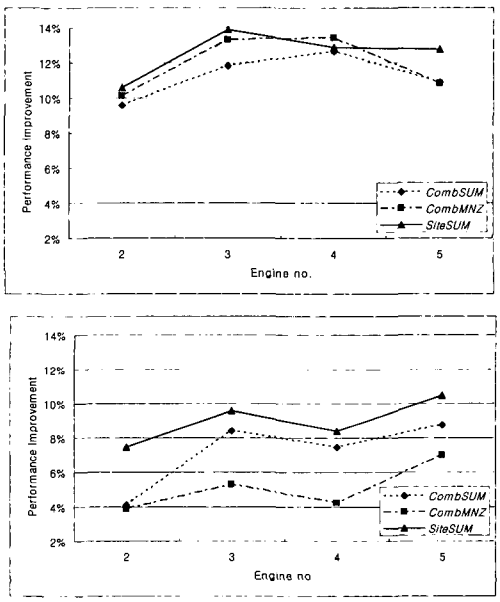
이전의 연구 결과에서처럼 내용 기반 검색에서는 공통적으로 나타나는 문서에 높은 유사도 값을 부여하는 *CombMNZ*가 효과적이었다. 하지만 엔트리 페이지 검색에서는 *CombSUM*보다도 낮은 성능을 보이고 있다. 그리고 엔트리 페이지 검색의 경우에는 검색 엔진의 결과를 결합하면 오히려 평균 정확도 향상치가 음수가 된다. 이것은 각 검색 엔진의 결과를 문서 단위로만 보는 기존의 함수로는 엔트리 페이지 검색 엔진의 성능을 올리는데 한계가 있음을 나타낸다. 이에 반해 중복 단위를 도메인 명으로 확장한 *SiteSUM*은 엔트리 페이지 검색에서 좋은 성능을 보임을 알 수 있다.

엔트리 페이지 검색의 결과에서 검색 엔진 개수가 2, 3개에서 매우 좋은 성능 향상을 보인다. 이는 참여한 엔진 중 내용 기반 검색 엔진 형태의 성능이 매우 저조한 엔진들간의 결합에 기인한다. 이러한 엔진들의 결과를 합칠 때 디렉터리별로 상위 디렉터리 결과를 선호한 결합 함수의 특성에 의해 매우 좋은 성능 향상을 보였다. 따라서 성능이 저조한 엔진들만의 결합 때 매우 좋은 성능 향상을 보였기 때문에 이러한 결과를 얻었다.

(그림 2)는 최고 성능 검색 엔진 선택 실험의 결과이다.




(그림 1) 무작위 선택 실험의 결과 - 내용 기반 검색(위), 엔트리 페이지 검색(아래)




(그림 2) 최고 성능 검색 엔진 선택 실험의 결과 - 내용 기반 검색(위), 엔트리 페이지 검색(아래)

무작위 선택 실험 보다는 좋은 성능을 보이지만 마찬가지로 *CombMNZ*가 엔트리 페이지 검색에서 좋은 성능을 보인다고 할 수 없다. 역시 *CombSUM*보다도 낮은 성능을 보이고 있다. 그리고 무작위 선택 실험과 같이 *SiteSUM*이 엔트리 페이지 검색에서 제일 좋은 성능 향상을 보이고 있다. 한편 *SiteSUM*이 내용 기반 검색에서도 좋은 성능을 보이고 있다. 이는 성능이 좋은 검색 엔진들이 하나의 도메인에서 여러 개의 결과를 제출하지 않으려는 특성이 있기 때문이다.

이와 같이 실험을 통해서 내용 기반 검색에서 유용한 메타 검색 방식이 엔트리 페이지 검색에서는 그렇지 않다는 것을 알 수 있다. 또한 엔트리 페이지 검색에서는 기존의

문서 단위 중복 외에도 도메인명 단위로 중복을 확장하는 것도 유용함을 알 수 있다. 즉 여러 검색 엔진의 결과가 정확한 엔트리 페이지로 일치하지 못한 경우에 기존 방법들에 비해 좋은 성능을 얻을 수 있음을 알 수 있다.

## 5. 상용 검색 엔진을 이용한 실험

본 절에서는 상용 검색 엔진에서의 *SiteSUM*의 효과를 보인다.

### 5.1 실험 환경

상용 검색 엔진으로는 사용자가 많이 사용하는 순서대로 *Naver*[2], *Daum*[3], *Yahoo*[4], *Empas*[5]를 사용한다[6]. 사용한 상용 검색 엔진의 특성은 웹 문서 검색과 사이트 검색 두 방식을 제공한다. 여기서 사이트 검색은 검색 엔진이 가지는 디렉터리를 검색하거나 따로 데이터베이스를 작성해서 검색하는 방식을 취한다. 웹 문서 검색 엔진은 내용 기반 검색이라고 볼 수 있고 사이트 검색 엔진은 엔트리 페이지 검색으로 볼 수 있다. *Daum*은 웹 문서 검색으로 *Google*[7]을 사용한다. 따라서 엔트리 페이지 검색은 *Daum*의 검색 결과를 그리고 내용 페이지 검색은 *Google*의 검색 결과를 사용한다[8].

내용 기반 검색을 위해서 100개의 질의를, 엔트리 페이지 검색을 위해서 120개의 질의를 사용하였다. 내용 기반 검색용 질의 100개 중 80개는 퀴즈 프로그램인 *I Love Quiz*[9]에서 추출하였으며 20개는 *ETRI-Kemong* 검색엔진 테스트용 질의 셋을 사용하였다[13]. 80개의 질의는 주로 *What*과 *Who*에 관련된 질의로 정답이 개체명 형태인데 반해 20개의 질의는 *Why*나 *How*에 관련된 질의로 문장이나 단락 형태의 정답을 가진다.

80개의 질의 중 예를 보이면 다음과 같다.

- *올 댓 재즈, 시카고, 뮤지컬, 제작자, 작곡자* - ***Bob Fosse***
- *스페인 화가, 초현실주의, 익살* - ***Salvador Dali***

20개의 질의의 예를 보이면 다음과 같다.

- *시민 혁명의 원인은?*
- *삼국의 통일 과정은?*

엔트리 페이지 검색을 위해서 사용한 120개의 질의 중105개의 질의는 잘 알려진 회사나 단체명을 사용하였으며, 15개의 질의는 사용자가 가고자 하는 곳을 설명하는 형태로 질의를 사용하였다. 105개의 단순 질의는 주로 도메인명만 가진 URL을 정답으로 가진다. 단순 질의의 예를 보이면 다음과 같다.

- *National Geographic* - ***http://www.nationalgeographic.co.kr***
- *JYP Entertainment* - ***http://www.jypent.co.kr***

15개의 복합 질의의 예를 보이면 다음과 같다.

- *대구대 불문학과* - ***http://french.taegu.ac.kr/~avecnous***
- *KAIST 최기선 교수님 홈페이지* - ***http://kibs.kaist.ac.kr/kschoi/***

본 연구에서는 내용 기반 검색 질의 두 종류를 *QUIZ, KEMONG*이라고 줄여서 나타내며, 엔트리 페이지 검색 용 단순 질의와 복합 질의는 *Simple, Complex*라고 줄여서 부른다.

### 5.2 상용 검색 엔진의 특성

<표 4>와 <표 5>는 각 상용 검색 엔진의 성능을 보인다. MRR과 %fail은 상위 10개의 문서를 이용해서 계산했다. 표에서 웹 문서 검색 엔진은 검색 사이트에서 제공하는 웹 문서 검색 결과 옵션을 선택하여 얻어낸 결과를 평가한 것이고, 사이트 검색 엔진은 검색 사이트에서 제공하는 사이트 검색 결과 옵션을 선택했을 때의 결과이다. 표에서 알 수 있듯이 내용 기반 검색에 대해서는 사이트 검색 엔진을 이용해서는 정답을 거의 찾아내지 못하고 있다. 반면 엔트리 페이지 검색에 대해서는 사이트 검색이 좋은 결과를 보이고 있음을 알 수 있다. 또한 엔트리 페이지 검색에 대해서는 웹 문서 검색 엔진으로도 어느 정도 좋은 결과를 보이고 있음을 알 수 있다. 이는 *inlink* 정보를 이용하는*Page Rank*나 URL의 깊이를 고려하는 *URL Depth*와 같은 정보를 사용하고 있기 때문으로 보인다[14]. 따라서 도메인명이 정답인 경우는 웹 문서 검색 엔진이 좋은 결과를 보이고 있다. 그러나 복합 질의인 경우에 대해서는 사이트 검색 엔진이 좋은 결과를 보이고 있다. 여기서 *Yahoo*는 웹 문서 검색 엔진의 결과로는 사이트의 결과를 거의 제공하고 있지 않음을 알 수 있다.

### 5.3 SiteSUM을 이용한 메타 검색

상용 검색 엔진의 결과를 합치기 위해서, 검색 엔진이 제공하는 각 문서에 대해서 순위에 기반해서 유사도를 추정한다(수식 8).

$$s(d) = \frac{1}{r(d)} \qquad \text{(수식 8)}$$

<표 6>과 <표 7>은 상용 검색 엔진의 결과를 합친 메타 검색의 성능을 보여준다.

앞에서 얘기된 바와 같이 내용 기반 검색에서는 *CombMNZ* 방식이 *CombSUM*보다 좋은 결과를 보임을 알 수

2) http://www.naver.com
3) http://www.daum.net
4) http://www.yahoo.co.kr
5) http://www.empas.com
6) 2003년 11월 http://rankey.com의 통계 자료에 기반하여 선정했다
7) http://www.google.co.kr
8) 본 실험은 2003년 11월에 시행되었다.
9) http://www.imbc.com/tv/ent/lovequiz/

〈표 4〉 내용 기반 검색 질의에 대한 상용 검색 엔진의 성능

| | 웹 문서 검색 엔진 | | | | | | 사이트 검색 엔진 | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Quiz | | | KEMONG | | | Quiz | | | KEMONG | | |
| | *P@1* | *P@5* | *P@10* | *P@1* | *P@5* | *P@10* | *P@1* | *P@5* | *P@10* | *P@1* | *P@5* | *P@10* |
| *Yahoo* | 0.47 | 0.27 | 0.18 | 0.25 | 0.18 | 0.14 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| *Empas* | 0.80 | 0.64 | 0.56 | 0.20 | 0.25 | 0.27 | 0.0 | 0.0 | 0.0 | 0.05 | 0.01 | 0.01 |
| *Google* | 0.54 | 0.36 | 0.30 | 0.25 | 0.23 | 0.16 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| *Naver* | 0.70 | 0.57 | 0.51 | 0.10 | 0.24 | 0.27 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |

〈표 5〉 엔트리 페이지 검색 질의에 대한 상용 검색 엔진의 성능

| | 웹 문서 검색 엔진 | | | | 사이트 검색 엔진 | | | |
|---|---|---|---|---|---|---|---|---|
| | Simple | | Complex | | Simple | | Complex | |
| | MRR | %fail | MRR | %fail | MRR | %fail | MRR | %fail |
| Yahoo | 0.04 | 93.3 | 0.02 | 93.8 | 0.49 | 50.0 | 0.19 | 81.3 |
| Empas | 0.68 | 30.8 | 0.13 | 87.5 | 0.43 | 56.7 | 0.25 | 75.0 |
| Google | 0.57 | 35.6 | 0.07 | 87.5 | 0.30 | 69.2 | 0.19 | 81.3 |
| Naver | 0.62 | 34.6 | 0.06 | 93.8 | 0.42 | 55.8 | 0.34 | 62.5 |

〈표 6〉 내용 검색 질의에 대한 메타 검색의 성능

| | 웹 문서 검색 엔진 | | | | | | 사이트 검색 엔진 | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Quiz | | | KEMONG | | | Quiz | | | KEMONG | | |
| | *P@1* | *P@5* | *P@10* | *P@1* | *P@5* | *P@10* | *P@1* | *P@5* | *P@10* | *P@1* | *P@5* | *P@10* |
| *CombSUM* | -9.4% | -6.3% | -2.9% | -20.0% | 16.0% | -13.0% | 0% | 100% | 100% | -100% | 0% | 100% |
| *CombMNZ* | -7.8% | -3.6% | -0.2% | 20.0% | 40.0% | -13.0% | 0% | 100% | 100% | -100% | 0% | 100% |
| *SiteSUM* | -25.0% | -7.3% | -3.6% | -40.0% | 12.0% | -13.0% | 0% | 100% | 100% | -100% | 0% | 100% |

〈표 7〉 엔트리 페이지 검색 질의에 대한 메타 검색의 성능

| | 웹 문서 검색 엔진 | | | | 사이트 검색 엔진 | | | |
|---|---|---|---|---|---|---|---|---|
| | *Simple* | | *Complex* | | *Simple* | | *Complex* | |
| | *MRR* | *%fail* | *MRR* | *%fail* | *MRR* | *%fail* | *MRR* | *%fail* |
| *CombSUM* | 13.6% | -59.4% | 20.2% | -7.1% | 25.4% | -28.8% | 1.1% | 0% |
| *CombMNZ* | 6.6% | -59.4% | 20.2% | -7.1% | 24.9% | -28.8% | -4.4% | 0% |
| *SiteSUM* | 25.3% | -59.4% | 29.8% | -14.3% | 25.4% | -28.8% | 1.1% | 0% |

〈표 8〉 웹 문서 검색 엔진과 사이트 검색 엔진 결과의 결합

| | 내용 기반 검색 질의 | | | | | | 엔트리 페이지 검색 질의 | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Quiz | | | KEMONG | | | Simple | | Complex | |
| | P@1 | P@5 | P@10 | P@1 | P@5 | P@10 | MRR | %fail | MRR | %fail |
| Y.SUM | 1.1% | -1.9% | 0.0% | 0.0% | 0.0% | 3.6% | -15.7% | -3.8% | -17.8% | -0.1% |
| Y.MNZ | 1.1% | -1.9% | 0.0% | 0.0% | 0.0% | 3.6% | -15.7% | -3.8% | -26.0% | -0.1% |
| Y.SITE | 1.1% | -1.1% | 0.0% | 0.0% | -5.6% | 0.0% | -0.2% | -3.8% | -17.8% | -0.1% |
| E.SUM | 0.0% | 0.8% | -0.7% | 0.0% | 4.0% | -3.7% | 2.8% | -15.7% | 0.0% | -8.3% |
| E.MNZ | 0.0% | 0.3% | -0.7% | 0.0% | 4.0% | -3.7% | 2.8% | -15.7% | 0.0% | -8.3% |
| E.SITE | 0.0% | 1.1% | -0.7% | 0.0% | 8.0% | -5.6% | 5.0% | -18.8% | 25.0% | -8.3% |
| G.SUM | -0.6% | 1.9% | 0.0% | 0.0% | 0.0% | 3.1% | 9.4% | -16.3% | -34.2% | -0.1% |
| G.MNZ | -0.6% | 1.9% | 0.0% | 0.0% | 0.0% | 3.1% | 10.5% | -16.3% | -42.4% | -0.1% |
| G.SITE | -2.8% | 1.9% | -1.0% | 0.0% | 0.0% | 3.1% | 10.7% | -16.3% | -23.2% | -0.1% |
| N.SUM | 0.0% | 0.9% | -0.8% | -50.0% | 0.0% | 0.0% | 16.3% | -27.7% | -32.0% | 0.0% |
| N.MNZ | 0.0% | 0.9% | -0.8% | -50.0% | 0.0% | 0.0% | 16.3% | -27.7% | -33.6% | 0.0% |
| N.SITE | 0.0% | 0.9% | -1.0% | -50.0% | -4.2% | -1.9% | 18.1% | -27.7% | -2.0% | 0.0% |
| A.SUM | -9.4% | -5.0% | -2.7% | 0.0% | 4.0% | -13.0% | 20.4% | -65.7% | -24.4% | 0.0% |
| A.MNZ | -7.8% | -3.1% | -0.5% | 50.0% | 40.0% | -16.7% | 17.4% | -62.5% | -30.8% | 0.0% |
| A.SITE | -29.8% | -7.8% | -4.5% | -25.0% | -4.0% | -14.8% | 27.3% | -65.7% | -17.3% | 0.0% |

있다. 그러나 메타 검색으로 인해 성능을 향상시키는 경우가 거의 드물다. 반면 엔트리 페이지 검색에 있어서는 *CombSUM* 방식이 근소하게 *CombMNZ* 보다 좋은 결과를 보임을 알 수 있다. 그리고 *SiteSUM* 방식이 엔트리 페이지 검색 질의에 대해서 제일 좋은 성능 향상을 보임을 알 수 있다. 그러나 사이트 검색 엔진의 결과에 대해서는 *CombSUM* 방식에 비해서 차이를 보이지 않고 있다. 이는 사이트 검색 엔진의 결과가 아직은 도메인명만 이루어진 URL이나 상업용으로 등록된 URL을 엔트리 페이지라고 간주하고 결과를 출력하는 상용 검색 엔진의 특성 때문이다.

표 8은 각 검색 엔진의 웹 문서 검색 결과와 사이트 검색 결과를 합쳤을 때의 성능을 보여준다.

여기에서 *'Y.SUM'*은 *yahoo*의 두 검색 결과를 *CombSUM*으로 합쳤을 때의 결과를 보여준다. 마찬가지로 *E, G, N, A*는 각각 *empas, google, naver, all*을 뜻한다. 마지막 *A*는 각 검색 엔진의 합친 결과들을 모두 합쳤을 때의 결과를 보여준다. 그리고 *'MNZ'*와 *'SITE'*는 각각 *CombMNZ*와 *SiteSUM*을 의미한다. 단순 질의의 경우에는 웹 문서 검색

결과와 사이트 검색 결과를 합쳐서 출력하는 것이 보다 좋은 결과를 얻을 수 있음을 알 수 있다. 즉 지금처럼 순차적으로 나열하는 방식보다 결합해서 내는 것이 더 좋을 수 있다는 것을 알 수 있다. 반면 복합 질의에 대해서는 아직 상용 검색 엔진의 수준이 좋지 않음을 알 수 있다.

## 6. 결 론

본 연구에서는 엔트리 페이지 검색의 결과를 결합하는 메타 검색 방법을 제시하였다. 내용 기반 검색 엔진의 결과를 결합할 때에는 여러 검색 엔진에 출현한 문서에 가중치를 두는 *CombMNZ*가 뛰어난 성능을 보인다고 알려져 있다. 하지만, 이 함수를 엔트리 페이지 검색의 결과를 결합하는 데 그대로 적용하면 좋은 성능을 얻지 못한다. 엔트리 페이지 검색은 내용 기반 검색에 비해 정답 문서의 개수가 훨씬 적기 때문에 여러 검색 엔진에 나타난 문서를 강조하는 내용 기반 검색에서 사용되는 메타 검색의 휴리스틱을 적용할 경우 좋은 결과를 얻기 어렵다.

본 연구에서는 엔트리 페이지 검색의 결과를 결합할 때에는, 여러 검색엔진에서 제공하는 문서뿐만 아니라 도메인명도 고려한다. 여러 검색 엔진에서 비록 동일한 문서는 제공하고 있지 않더라도 동일한 도메인의 결과를 상위 결과로 제시할 경우 이는 해당 도메인의 결과를 중요하게 생각할 수 있는 정보가 된다. 본 연구에서는 문서 단위로 중요도를 계산할 뿐만 아니라 이를 도메인 단위로도 확장하여 이를 결합한 *SiteSUM*을 제안했다. 이 모델을 사용하면 사이트 단위의 중요도에서 좋은 점수를 얻은 사이트에 대해 문서 단위의 점수를 이용해 알맞은 엔트리 페이지를 찾는 것이 가능하다. TREC의 결과물을 통한 실험과 상용 검색 엔진 결과물을 이용한 실험을 통해서, 최근 대두되고 있는 엔트리 페이지 검색을 위해서는 내용 기반 검색과는 다른 형태의 결합 함수가 필요함을 알 수 있었다. 그리고 본 연구에서 제안하는 *SiteSUM*이 좋은 대안이 될 수 있음을 알 수 있었다. *SiteSUM*은 현재 서비스 되고 있는 상용 검색 엔진에도 바로 적용할 수 있음을 알 수 있었다.

앞으로의 연구로는 새롭게 관심이 대두되고 있는 서비스 검색을 위한 메타 검색에 대한 연구가 필요하다. 아울러 현재와 같이 사용자가 내용 기반 검색인지 엔트리 페이지 검색인지 지정해 주는 방식이나 두 가지 결과를 모두 제시하는 방식에서 자동으로 추정하는 연구 또한 필요하다.

## 참 고 문 헌


[1] Baeza-Yates, R., and Ribeiro-Neto, B. "Modern Information Retrieval", Essex England: Addison-Wesley Pub Co, 1999.

[2] Frakes, W.B., Baeza-Yates, R. "Information Retrieval Data Structures & Algorithms", Prentice Hall Inc., Englewood Cliffs, New Jersey 1992.

[3] Shaw, J., Fox, E. "Combination of Multiple Searches", In Text REtrieval Conference (TREC-3), Gaithersburg, Maryland, pp. 105-108, 1994.

[4] Bartell, B.T., Cottrell, G.W., Belew, R.K. "Automatic Combination of Multiple Ranked Retrieval Systems", In Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Dublin, Ireland, pp. 173-181, 1994.

[5] Dreilinger, D., Howe, A.E. "Experiences with Selecting Search Engines using MetaSearch", ACM Transactions on Information Systems, vol. 15, pp. 195-222, 1997.

[6] Lee, J.H. "Analyses of Multiple Evidence Combination", In Proceedings of the 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Philadelphia, pp. 267-276, 1997.

[7] Lee, J.H. "Combining Multiple Evidence from Different Properties of Weighting Schemes", In Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Seattle, Washington, pp.180-188, 1995.

[8] Aslam, J., Montague, M. "Models for MetaSearch", In Proceedings of the 24th Annual International ACM SIGIR conference on Research and Development in Information Retrieval, New Orleans, LA, pp. 267-284, 2001.

[9] Montague, M. "MetaSearch: Data Fusion for Document Retrieval", PhD dissertation, Dartmouth College, 2002.

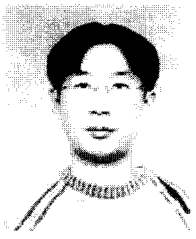
[10] Broder, A. "A Taxonomy of Web Search", SIGIR Forum, 36(2), 2002.

[11] Manmatha, R., Rath, T., Feng, F. "Modeling Score Distributions for Combining the Outputs of Search Engines", In Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, New Orleans, LA , pp. 267-275, 2001.

[12] Harman, D. "Overview of the Third Text REtrieval Conference", In Text REtrieval Conference (TREC-3), Gaithersburg, Maryland pp. 1-20, 1994.

[13] KEMONG "The Kemong Company new Encyclopedia", Kemong Corp., Seoul: KEMONGSA Publishing Co. 1992.

[14] Westerveld, T., Kraaij, W., and Hiemstra, D. "Retrieving Web pages using Content, Links, Urls and Anchors" In *Text REtrieval Conference*(TREC-10) (pp. 663-672). Gaithersburg, Maryland, 2001


**강 인 호**

e-mail : inho97.kang@samsung.com

1997년 경북대학교 컴퓨터공학과 졸업(학사)

1999년 KAIST 전산학과(공학석사)

2004년 KAIST 전산학과(공학박사)

2004년~현재 삼성종합기술원 Computing LAB 전문연구원

관심분야 : 정보검색, 정보추출, 한국어 정보처리