http://dx.doi.org/10.3745/KIPSTB.2011.18B.4.231

페이지 랭크지수와 질의 확장을 이용한 재랭킹 방법

김 태 환[†] · 전 호 철[††] · 최 중 민[†††]

요 약

사람들은 월드 와이드 웹 상에서 사용자가 원하는 정보를 검색하는 여러 알고리즘들을 구현해 왔다. 이렇게 구현된 검색 알고리즘 중 가장 좋은 기술을 가지고 있는 곳은 페이지랭크(PageRank)방식의 구글이다. 하지만 외부에서 참조하는 링크가 많은 문서를 가지고 있는 문서 즉, 대중들이 관심을 가지는 문서를 상위에 보여주는 페이지랭크 방식으론 사용자가 원하는 문서를 찾아서 제공하지 못할 수 있다. 개인에게 가치가 있는 문서를 찾기보다 대중에게 가치가 있는 문서를 찾기 때문이다. 이러한 문제를 해결하기 위하여 본 논문에서는 어휘의 의미를 정확히 표현하고 있는 워드넷을 이용하여 사용자 질의 이력 정보를 분석하여 현재 질의를 확장한 개인적 가치와 페이지 랭크지수를 이용한 대중적 가치를 모두 고려한 방법을 제안한다. 실험결과 제안한 방법은 상위 30개의 검색결과 중 평균 약 60% 결과들에 대해 만족하는 것으로 나타났으며, 구글 검색 결과에 비해 평균 약 14% 향상된 만족도를 나타내었다.

키워드 : 워드넷, 개인화, 정보 검색, 페이지 랭크

A Reranking Method Using Query Expansion and PageRank Check

Tae-Hwan Kim[†] · Ho-Chul Jeon[††] · Joong-Min Choi[†††]

ABSTRACT

Many search algorithms have been implemented by many researchers on the world wide web. One of the best algorithms is Google using PageRank technology. PageRank approach computes the number of inlink of each documents then ranks documents in the order of inlink members. But it is difficult to find the results that user needs, because this method find documents not valueable for a person but valueable for the public. To solve this problem, We use the WordNet for analysis of the user's query history. This paper proposes a personalized search engine using the user's query history and PageRank Check. We compared the performance of the proposed approaches with google search results in the top 30. As a result, the average of the r-precision for the proposed approaches is about 60% and it is better as about 14%.

Keywords : WordNet, Personalized, Information Retrieval, PageRank

1. 서 론

월드 와이드 웹의 사용자가 폭발적으로 증가함에 따라 대량의 정보가 개인, 기업 홍보 또는 상업적인 목적으로 생성되고 있다. 현재 국내에는 수백 만개 수준의 웹 문서가, 세계적으로는 억 단위 수준의 웹 문서가 산재해 있으며 그 수는 빠르게 증가하고 있다[1]. 이와 같이 방대한 정보는 검색 엔진을 통해 검색 결과의 순위로 나타나게 된다. 하지만 현재의 검색 엔진은 개인에게 적합한 정보를 제공하기보다 다수에게 적합한 정보를 보여주고 있기 때문에 사용자는 자신에게 적합한 정보를 찾기 위하여 많은 웹 문서를 방문하거나, 질의어를 확장하거나 새로운 질의어를 입력해야 한다.

통계적으로 사용자는 평균 2.21개의 질의어를 입력하고, 그 중 58%의 사용자는 첫 번째 검색 결과 페이지 내에서 자신에게 적합한 정보를 찾는다. 만약 첫 번째 검색 페이지 내에 자신에게 적합한 페이지가 없다면 다른 질의어를 입력하거나 질의어를 확장하여 다시 검색한다[2]. 이처럼 사용자가 평균적으로 입력하는 질의어만으로 사용자에게 적합한 문서를 찾기 어렵고 또한 적합한 문서를 찾는데 많은 시간이 소요된다. 이러한 사용자의 불편과 적합한 정보를 찾는데 소요되는 비용을 줄이기 위하여 검색 결과의 순위 재조정에 대한 연구가 진행되고 있다. 검색 결과의 순위 재조정에 대한 연구는 개인화 기반 순위 재조정 방법과 대중화 기반 순위 재조정 방법이 있다.

※ 본 논문은 지식경제부 산업원천기술개발사업(10035348, 모바일 플랫폼 기반 계획 및 학습 인지 모델 프레임 워크 기술 개발)의 지원으로 수행 되었음.
† 준 회 원 : 한양대학교 컴퓨터공학과 박사과정
†† 준 회 원 : 한양대학교 컴퓨터공학과 박사
††† 정 회 원 : 한양대학교 컴퓨터공학과 교수
논문접수 : 2011년 1월 12일
수 정 일 : 1차 2011년 3월 14일, 2차 2011년 4월 20일
심사완료 : 2011년 5월 2일

대중화 기반 순위 재조정 방법의 대표적인 알고리즘은 구글에서 사용하는 PageRank알고리즘으로 웹 문서의 구조적 특징인 하이퍼링크(Hyperlinks)를 이용하여 외부에서 참조하는 링크가 많은 웹 문서는 중요한 문서라 판단하여 순위를 조정하는 알고리즘이다[3]. 이러한 PageRank의 구조적 특징으로 인하여 개인 사용자에게 적합한 문서보다는 대중적으로 적합한 문서를 검색 결과의 상위에 위치시킨다. 하지만 오늘날 많은 상업적인 목적을 가진 사이트들은 PageRank의 알고리즘을 악용하여 자신의 사이트를 검색 상위에 위치시키면서 랭킹의 정확도와 공정성에 위협하고 있다[4].

개인화 기반 순위 재조정은 사용자의 이전 질의어 및 사용자가 방문했던 문서 등을 기록한 정보를 분석하여 순위를 재조정하는 방법[5, 6]과 사용자와 비슷한 관심거리 또는 주제를 공유하는 그룹 즉, 커뮤니티의 사용자들과의 협업된 정보를 이용하여 순위를 재조정하는 방법[7], 구글의 PageRank를 변형한 방법[8, 9, 10]등이 있다.

본 논문에서는 검색에서의 사용자 불편과 적합한 정보를 찾는데 소요되는 비용을 줄이기 위하여 대중적 가치와 개인적 가치를 혼합한 개인화 검색 엔진을 제안한다. 여기서 대중적 가치란 월드와이드웹 상에 외부에서 참조하고 있는 링크가 많은 문서는 인터넷을 사용하는 사람에게 유용한 정보를 담고 있는 문서를 말한다. 개인적 가치란 인간의 어휘지식을 모방하여 의미를 최대한 정확히 표현한 워드넷을 이용하여 사용자의 현재 질의어와 이전에 사용한 질의어를 상호 매칭시켜 질의어와 의미적으로 연계된 다른 용어를 질의어에 추가로 확장한 질의어를 말한다.

본 논문의 구성은 다음과 같다. 먼저 2장에서는 정보 검색의 순위화 필요성 및 개인화 검색 결과의 순위를 재조정하는 방법에 대하여 설명한다. 3장에서는 본 논문에서 제시하고 있는 시스템의 구조와 개인화 검색 방법에 대해 설명한다. 4장에서는 제안하고 있는 방법을 실험 평가하기 위한 시나리오를 작성하고 5장에서는 현재 구글에서 검색된 결과와 본 논문에서 제시한 순위를 재조정한 결과를 비교 평가한다. 6장에서 결론 및 향후 연구를 기술한다.

2. 관련 연구

2.1 정보 검색 결과의 재랭킹 필요성

정보검색은 수집된 문서를 분석하여 저장하고, 저장된 파일로부터 사용자의 요구에 적합한 결과를 탐색하여 제공하는데 의의가 있다. 이러한 정보검색은 질의의 형태로 표현되는데 사용자의 질의어를 반영하여 저장된 문서를 분석한 후 이를 비교하여 관련성이 높은 문서 순으로 정렬하는데 이를 위해 각 문서에 점수를 할당한다. 이때 사용되는 수식이 검색 모델이며 정렬된 문서는 순위화 되었다고 한다. 이렇게 순위화된 문서는 같은 질의어를 사용한 모든 사용자에게 같은 결과로 나타난다. 하지만 사용하는 질의어가 같더라도 사용자가 원하는 문서가 다르기 때문에 입력하는 질의어만으로 사용자에게 적합한 문서를 찾기 어렵고 또한 적합한 문서를 찾는데 많은 시간이 소요된다. 이러한 사용자의 불편과 적합한 정보를 찾는데 소요되는 비용을 줄이기 위하여 검색 결과의 순위 재조정에 대한 연구가 필요하다. 즉, 검색 사용자는 모든 검색 결과를 참조하는 것이 아니고 상위에 개제된 몇 개의 문서만을 보기 때문에 사용자가 원하는 문서를 검색 결과의 상위로 재랭킹하는 연구가 필요하다.

2.2 개인화된 검색 결과의 재랭킹 관련 연구

PageRank를 수정하여 개인화된 검색 결과의 재랭킹 관련 연구는 Personalized PageRank 방법, Topic-Sensitive-PageRank 방법, Topic-Sensitive-PageRank와 개인화를 결합한 방법 등이 있다.

Topic-Sensitive-PageRank[8]는 수집된 문서를 주제어와 PageRank의 지수를 계산하여 주제별 문서의 순위를 구한다. 사용자의 질의어가 입력되면 질의어와 주제별 문서의 순위를 비교하여 수치화 하고, ODP[11]의 16개의 카테고리를 이용하여 카테고리별 수치화된 문서를 순위화하여 재랭킹하는 알고리즘을 제안한다. 하지만 사용자 질의어의 하위 주제어를 선택할 때 확률을 기반으로 하고 있기 때문에 주제어가 편향되어 선택될 수 있다. 예를 들어 질의어 table tennis를 살펴보면 sports를 하위주제어로 가질 확률은 0.53이고 shopping을 하위주제어로 가질 확률은 0.14이므로 질의어 table tennis에 대한 서브 주제어는 항상 sports로 편향되어 선택되어 진다. 이러한 편향된 하위 주제어 선택과 16개의 정해진 카테고리로 인해 사용자에게 중요한 문서로 재랭킹하기 보다 질의어와 주제, 질의어와 문서, 질의어와 카테고리의 관계를 더 중요시하여 재랭킹하였다.

Personalized PageRank[9]는 대중적 가치를 기반으로 순위를 조정하는 PageRank에 개인화 개념을 추가한 알고리즘이다. 각 사용자는 관심 있는 문서의 집합을 선택한다. 선택된 문서의 집합들 중 의미 단어를 추출하여 Personalized PageRank Vector(PPV)를 구성한다. 구성된 PageRank Vector를 기반으로 검색 결과를 재랭킹하는 알고리즘을 제안한다. 하지만 사용자에 의해 선택된 문서에서 관심어를 추출하여 벡터로 구성하는 것은 사용자가 원하지 않는 단어를 추출할 수 있기 때문에 많은 에러를 포함한다. 이러한 에러는 재랭킹된 검색 순위에 영향을 미친다.

Topic-Sensitive-PageRank와 개인화를 결합한 방법[10]은 사용자에 의해 선택된 문서를 분석하여 초기의 사용자 프로파일로 설정하고, 프로파일을 학습하여 topic-sensitive-pagerank에 적용하여 검색 결과를 재랭킹하는 알고리즘을 제안하였다. 하지만 사용자에 의해 선택된 문서에서 사용자 프로파일을 설정하고 학습하는 것은 사용자가 원하지 않는 단어를 포함할 수 있기 때문에 많은 에러를 포함하며, 이러한 에러는 재랭킹된 검색 순위에 영향을 미친다.

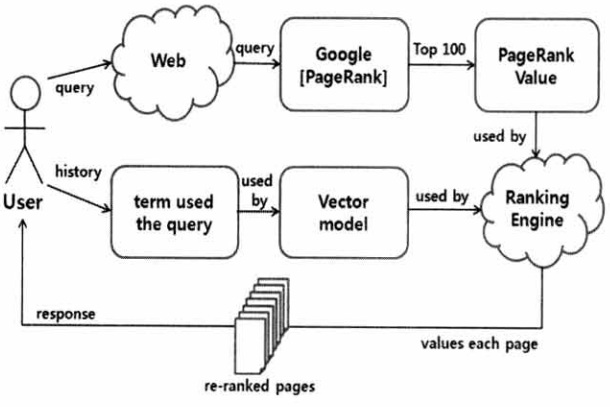
본 논문에서 제안하는 방법은 대중적 가치를 기반으로 순위를 조정하는 PageRank와 어휘의 의미를 정확히 표현하고 있는 워드넷을 이용하여 사용자 질의 이력 정보를 분석한

벡터 모델 값을 결합하여 검색 결과를 재랭킹하는 방법을 제안한다.

3. 시스템 구조와 개인화 검색 방법

3.1 시스템 구조

본 논문에서 사용하고 있는 시스템의 구조는 (그림 1)과 같은 구조로 이루어져 있다.



(그림 1) 개인화 정보 검색 구조

시스템 구조는 크게 3가지로 분류할 수 있다. 첫 번째 단계는페이지 랭크 지수를 구하는 단계로 사용자의 질의에 해당하는 검색 결과 100개의 사이트에 대해서 각각의 페이지 랭크 지수를 구한다. 두 번째 단계는 벡터 모델 값을 구하는 단계로 이전에 사용된 사용자의 질의를 이용하여 벡터 모델 값을 구한다. 세 번째 단계는 페이지 랭크 지수와 벡터 지수를 결합한 단계로 이전에 나열되었던 목록을 페이지 랭크 지수와 벡터 모델 값을 결합하여 제시한 순위를 재조정 한다.

3.1.1 페이지 랭크 지수

페이지 랭크 알고리즘은 기본적으로 인링크(자신을 향하는 링크)의 개수와 인링크 페이지의 가중치를 기준으로 모든 웹 페이지들에 페이지 랭크 지수를 부여한다. 페이지 랭크 지수의 가장 큰 특징은 질의어 없이 웹 페이지의 랭크 지수를 구할 수 있다는 것이다. 하지만 페이지 랭크를 이용해 실험하는 연구자에게 있어서 실제 존재하는 모든 문서의 페이지 랭크 지수를 알 수 없기 때문에 이를 이용한 실험에 한계가 있다. 우리는 이러한 한계 때문에 페이지 랭크 지수를 이용한 검색 사이트인 구글에서 사용자 질의를 하여 상위 100개의 사이트를 크롤링하고 크롤링된 사이트 각각의 랭크지수를 구하였다.

수식 1은 페이지 랭크 지수를 구하는 공식이다. $PR(A)$는 문서 A의 페이지 랭크, $PR(t_1)$는 문서 A를 가리키는 문서 t_1의 페이지 랭크, $C(t_1)$는 문서 t_1가 가리키는 문서 개수, d는 사용자가 특정 문서에서 만족하지 못하고 다른 문서로 이동할 확률을 나타낸다.

$$PR(A) = 1-d(1-\frac{RP(t_1)}{C(t_1)}-\frac{RP(t_2)}{C(t_2)}-\ldots-\frac{RP(t_N)}{C(t_N)}) \quad (1)$$

구글의 인링크 수와 페이지 링크 지수는 popuri[1] 에서 제공하는데 0부터 10사이의 페이지 랭크 지수를 반환하여 준다. 이 값을 1로 정규화 시키기 위하여 10으로 나누어 랭크 지수를 구하였다.

3.1.2 벡터 모델 값

벡터 공간 모델은 텍스트 문서를 식별자들의 벡터로 나타내는 대수적인 모델이다. 사용자 질의어에 대한 상위 100개의 사이트와 사용자가 이전에 사용한 검색 질의어는 벡터로 표현되며, 각각의 차원은 개별 단어에 대응된다. 만약, 문서 내에 특정 단어가 포함되어 있다면, 벡터 내에서 해당 차원은 0이 아닌 값을 가지게 되는데 이것을 단어 가중치라고 한다[12]. 단어 가중치를 구하는 가장 잘 알려진 방법은 *tf-idf* 가중치를 구하는 방법이 있지만 우리는 검색 결과에 대한 상위 100개의 사이트를 이용한 순위를 재조정하는 방법을 제안하고 있기 때문에 *tf-idf*가 아닌 *tf-(1-idf)*를 이용한다.

수식 2는 *tf-(1-idf)*를 이용하여 수식 4의 $w_{i,j}$ 즉 단어의 가중치를 구하는 공식이다.

사용자 질의어는 현재 질의어와 유사한 의미를 가지고 있는 이전에 검색한 질의어를 질의어에 포함 시켜 구성한다. 수식 2에 의해서 단어의 가중치 w_{ij}를 구한 다음에 구성된 질의어와 상위 100개의 문서 사이의 내적 값을 각각 구한다.

$$w_{i,j} = f_{i,j} \times (1-\log\frac{N}{n_i}) \quad (2)$$

수식 3은 수식 2의 $f_{i,j}$ 값을 구하는 공식이다.

$$f_{i,j} = \frac{freq_{i,j}}{\max_l freq_{l,j}} \quad (3)$$

수식 4는 새로운 질의어와 검색과 관련된 이전에 사용된 질의어를 가지고 상위 100개의 사이트와 유사도를 구하는 벡터 공간 모델의 공식이다.

$$sim(d_j,q) = \frac{\vec{d_j}\bullet\vec{q}}{|\vec{d_j}|\times|\vec{q}|} = \frac{\sum_{i=1}^{t} w_{i,j}\times w_{i,q}}{\sqrt{\sum_{i=1}^{t} w_{i,j}^2}\times\sqrt{\sum_{j=1}^{t} w_{i,q}^2}} \quad (4)$$

1) http://popuri.us/

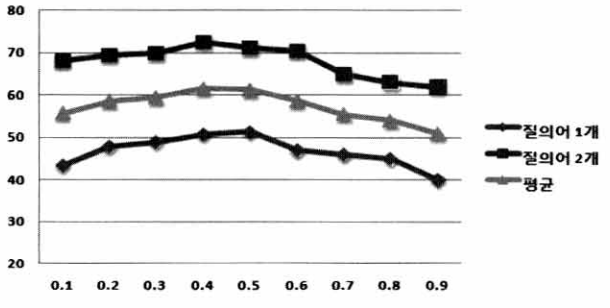
3.1.3 페이지 랭크 지수와 벡터 모델 값의 결합

두 개의 값을 결합할 때 1로 정규화 하기 위하여 페이지 랭크 값에 가중치 v를 곱하고 벡터 모델 값에는 (1-v)를 곱하였다. 가중치 v를 사용하는 의도는 페이지 랭크 값과 벡터 모델 값 중 어느 쪽에 비중을 더 주어야 성능이 향상되는지 알아보기 위함이다.

수식 5는 벡터 모델과 페이지 랭크 값을 결합한 공식이다. v는 0에서 1사이의 소수를 가지며 실험을 해본 결과 벡터 모델에 좀 더 가중치를 부여한 경우가 더 성능이 좋았다.

$$sim(d_j, q) = v(PR(A)) + (1-v)\frac{\sum_{i=1}^{t} w_{i,j} \times w_{i,q}}{\sqrt{\sum_{i=1}^{t} w_{i,j}^2 \times \sum_{j=1}^{t} w_{i,q}^2}} \quad (5)$$

이때, 수식 5에 해당하는 가중치 v를 0.1부터 0.9까지 다양하게 설정하여 질의어 1개일 때와 질의어 2개일 때의 성능을 측정한 실험 결과는 아래 (그림 2)와 같다. 이 결과에서 보면 가중치가 0.4일 때 가장 성능이 좋은 것을 알 수 있었으며, 이 결과를 토대로 검색 성능 측정을 위한 가중치는 0.4로 설정하였다. 이렇게 계산된 값을 이용하여 높은 값을 상위에 낮은 값을 하위로 편성하여 페이지의 순위를 재구성하고 이를 사용자에게 반환해 준다.

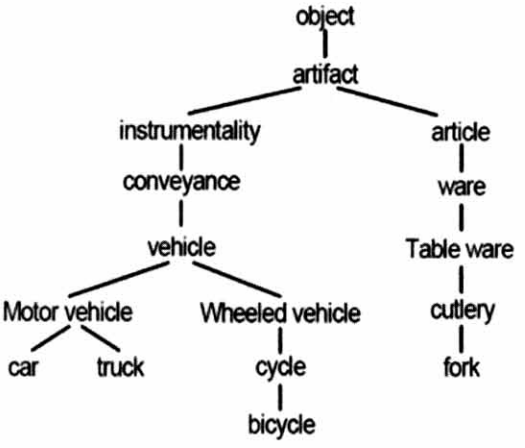


(그림 2) 가중치 값의 변화에 따른 정확률

3.2 사용자 질의어 구성

사용자가 이전 검색에서 사용한 질의어의 집합은 개인이 필요한 정보를 찾기 위해 사용된 단어이기 때문에 개인적 가치라 한다. 이전 검색에서 사용된 질의어는 워드넷을 이용하여 개념 별로 분류 한다. 이때 분류에 사용되는 워드넷의 특성은 카테고리 정보를 나타내는 synset_offset과 단어의 의미 종류를 나타내는 sense 이다.

전자의 synset_offset은 8자리로 구성되는데 그 값은 트리상의 상위 노드의 단어일수록 앞의 숫자가 0으로 채워진다. 예를 들어, (그림 3)은 워드넷에 표기된 단어의 계층적 구조를 나타내는 것으로 하위어 car의 synset_offset의 값은 02929975 이고, car의 최상위 부모 노드인 object의 synset_offset의 값은 00003122이고, object의 하위 노드인 artifact의 synset_offset의 값 또한 00020846 이다. 의미가 명확한 경우 synset_offset의 앞의 자리에 0이 한 개 이하로 나타나고 의미가 폭 넓게 사용하는 단어는 synset_offset의 앞의 자리에 0이 3개 이상 나타난다. 본 논문에서는 이러한 특성을 고려하여 synset_offset의 앞의 자리가 3개 이상 나타난 단어에 대해서는 질의 확장을 고려하지 않는다.



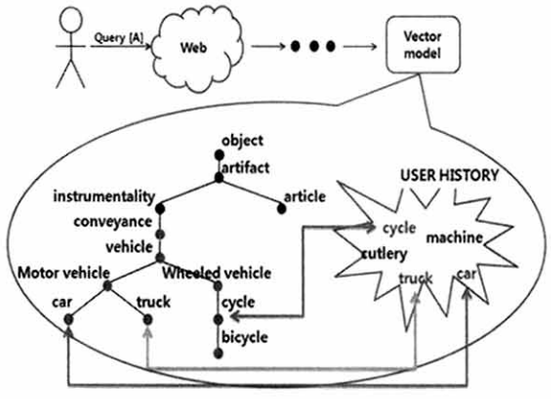
(그림 3) 워드넷에 표기된 단어의 계층적 구조

후자의 sense는 단어가 가지는 의미 정보를 나타낸다. 예를 들어, (그림 3)의 최하위 노드 car 인 경우 5개의 다른 의미를 가지기 때문에 sense는 #5 로 표기하며 각각의 의미는 #1 은 4개의 바퀴를 가진 자동차 #2는 철도위를 움직이는 차량 #3는 여객, 화물을 운송하는 차량을 의미한다. 이러한 의미 정보 sense은 car의 상위어인 motor vehicle에도 나타난다. motor vehicle은 1개의 의미만을 가지고 있고 의미는 car의 첫 번째 의미 #1과 같다. 이러한 의미적 특성은 질의어 확장에 사용한다.

다음은 위에서 설명한 2가지 특성을 고려하여 질의어를 확장하는 방법에 대한 설명이다. 예를 들어, 이전에 질의어로 사용된 단어 car와 bicycle 두 개의 단어가 있다고 한다면 car라는 단어의 상위어 는 motor vehicle, vehicle, conveyance, instrumentality, artifact, object 순으로 가지고, 마찬 가지로 bicycle의 상위어는 cycle, wheeled vehicle, vehicle, conveyance, instrumentality, artifact, object 순으로 가진다. 여기서 car와 bicycle 두 개의 상위어로 만나는 단어는 vehicle, conveyance, instrumentally, artifact, object인데 artifact, object는 synset_offset의 앞의 자리에 0이 3개 이상 나타나기 때문에 확장어 후보에서 제외한다. 확장 후보 단어 vehicle, conveyance, instrumentally와 질의어 car, bicycle의 의미 sense를 비교한다. car는 #1, #2, #3, #4, #5의 의미를 가지고, bicycle는 #1의 의미를 가지고, vehicle는 #1, #2, #3, #4의 의미를 가지고, conveyance는 #1, #2, #3, #4, #5의 의미를 가지고, instrumentality는 #1, #2, #3의 의미를 가진다. 5개의 단어 모두 #1의 의미를 가지기 때문에 #1의 의미적 연관성을 가진 단어라 판단할 수 있다. 이렇게 확장 후보 단어가 결정되면 사용자 이력 정보에 확장 후보 단어가 사용되었으면 질의어를 확장한다.

3.2.1 사용자의 질의어가 1개 일 때 구성 예

(그림 4)는 벡터 모델에서 이전에 사용된 질의어를 사용하는 방법의 한 예이다. 예를 들어, 사용자가 정보를 검색하기 위하여 "conveyance"를 질의어로 입력하면 워드넷에서 질의어 "conveyance"의 하위 집합은 {vehicle, Motor vehicle, Wheeled vehicle, car, truck, cycle, bicycle}이다. 하위 집합의 단어는 conveyance의 하위 단어로 모두 synset_offset의 앞의 자리에 0이 한 개 이하로 나타난다. 또한 7개의 단어 모두 #1의 의미를 가지기 때문에 #1의 의미적 연관성을 가진 단어라 판단하여 확장 후보 단어로 구성한다. 이 단어들은 [그림 4]에서 나타나고 있는 사용자의 이력정보{car, cycle, truck, cutlery, machine}와 일치하는 단어를 찾아낸다. 여기서 "conveyance"의 하위어로 일치하는 단어는 {car, cycle, truck}이다. 사용자의 질의어 "conveyance"에 이전 검색에 이용된 단어 중 conveyance의 하위어 {car, cycle, truck}을 포함시켜 검색된 페이지에 벡터 모델 값을 계산한다. 이렇게 계산된 값은 popuri 에서 제공되는 페이지 랭크 값과 결합하여 순위를 재조정 한다.



(그림 4) 이전 질의어를 이용하는 방법의 예

3.2.2 사용자의 질의어가 2개 이상일 때 구성 예

사용자의 질의어가 2개 이상일 때 두 개 이상의 질의어가 가지는 상위어가 워드넷 상에 존재할 때와 존재하지 않을 때 두 가지로 구분할 수 있다. 워드넷 상에 존재하지 않는 경우는 질의어 각각을 3.2.1절에서 이야기했던 방식으로 질의어 벡터를 구성하여 검색한다. 2개 이상의 질의어에 대한 상위어가 존재한다면 3.2에서 설명한 것과 같이 상위어의 하위 단어에서 synset_offset과 sense를 고려하여 확장 후보 단어를 구성하고, 사용자의 히스토리 집합과 일치하는 단어를 찾아내어 질의어를 확장하여 벡터로 구성하여 검색한다.

3.2.3 질의어 확장 알고리즘

3.2.1절의 확장 알고리즘은 [알고리즘 1]과 같다.

[알고리즘 1]에서 Userhistory는 사용자가 이전에 검색한 질의어의 집합이다. setDescendingOrder() 함수는 이전 질의어를 내림차순으로 정렬해 주는 함수이다. getSubTree() 함수는 워드넷에 존재하는 트리 중 현재 사용하는 질의어의

Algorithm 1. QUERY EXPANSION

```
1:  procedure VOID QUERYEXPANSION(query, Userhistory)
2:      historyList ← setDescendingOrder(Userhistory)
3:      queryExpansion ← query
4:      targetList ← getSubTree(query)
5:      while historyList is not Empty do
6:          compList ← getNextElement(historyList)
7:          if compList ⊆ targetList then
8:              queryExpansion.add(compList)
9:          end if
10:     end while
11: end procedure
```

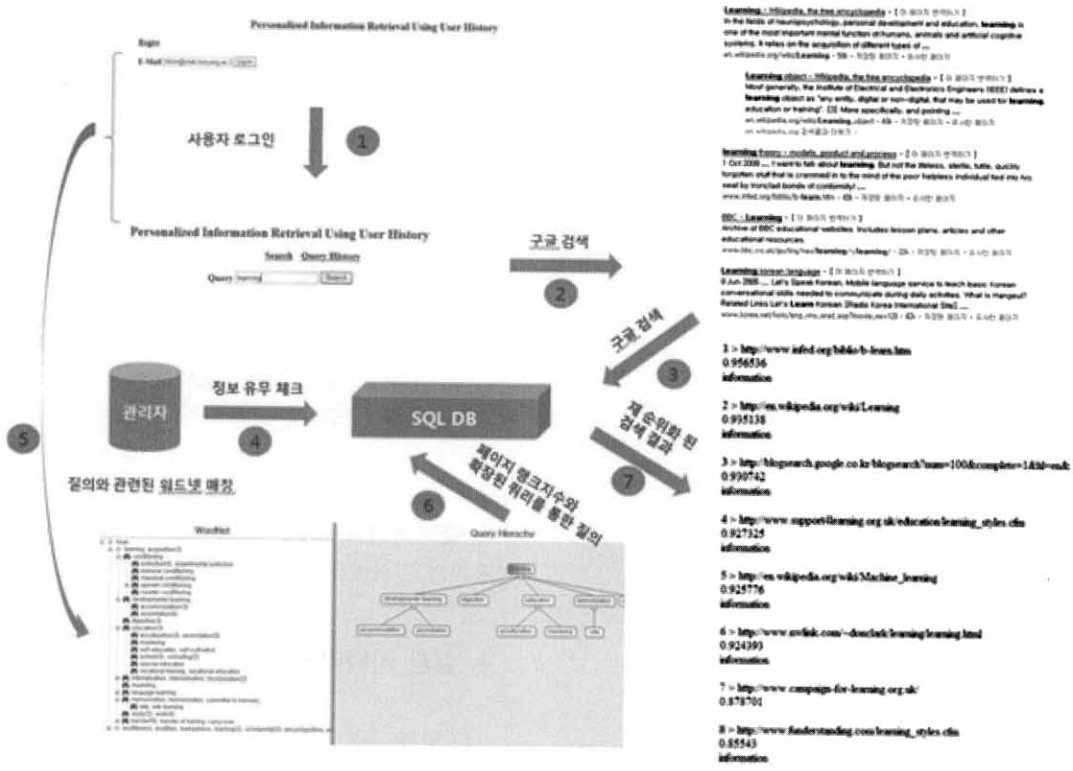
[알고리즘 1] 사용자 질의어 확장 알고리즘

하위어를 모두 가져오는 함수이다. [알고리즘 1]의 5-10 라인은 사용자의 이력정보에 나타난 단어가 워드넷에서 가져온 하위어에 포함되었는지 여부를 확인하여 포함되어있으면 사용자의 질의어를 추가하여 확장한다.

4. 실험 시나리오

시스템의 실험을 위한 시나리오는 (그림 5)와 같다. 검색을 이용하는 사용자는 자신의 이전 질의어를 사용하여 질의어를 확장하기 위하여 자신의 메일 계정으로 로그인 한다. 이는 개인의 정보를 질의어로 제한함으로써 개인의 정보를 제공하거나 수집을 허락하는 것에 대해 사용자들의 반감을 최소화 하기 위함이다. 로그인 후 (그림 5)의 과정 1과 같이 제공하는 사용자 인터페이스에 질의어를 입력하게 된다. 이때 시스템은 두 가지 일을 동시에 수행하게 되는데 하나는 검색에 이용된 질의어를 가지고 구글에서 검색하여 상위 100개의 사이트에 대해 구글에서 제공하는 페이지 랭크 지수 및 그 사이트가 해당 질의어에 대해 정보인지 정보가 아닌지 전문가가 체크하여 SQL DB에 저장하는 과정 2와 3을 진행한다. 다른 한편으로는 사용자가 질의한 질의어와 사용자가 이전에 질의한 질의어를 워드넷의 상위어와 하위어 관계를 이용하여 3장에서 기술한 것과 같이 질의어를 과정 5처럼 확장할 수 있다. 이렇게 확장된 질의어를 가지고 저장된 상위 100개의 검색 결과 순위에 다시 질의를 하여 저장된 페이지 랭크 지수와 확장된 질의어 벡터를 이용하여 순위를 재조정 하는 과정 6을 거치게 된다. 과정 7은 순위가 재조정된 결과 화면이다. 결과 화면은 사이트 URL과 사이트에 대해 수식 5를 이용하여 구한 값과 해당 사이트가 정보인지 아닌지를 나타낸다. 이렇게 출력된 결과에서 상위 30개에 대한 정확률을 구한다. 여기서 검색 정확률은 정답 셋에 있는 정답 개수만큼을 검색결과로 고려하는 R-Precision을 사용하여 측정하였다. R-Precision을 나타내는 수식은 다음과 같다.

$$RPrecision = \frac{\text{검색된정보의수}}{\text{적합한정보의수}} \times 100 \qquad (6)$$



(그림 5) 재랭킹 과정 및 실험 시나리오

5. 실험 방법 및 결과 분석

정보 검색이나 추론에 관련된 연구를 진행하는 연구실에서 사용하는 일반적인 단어를 구글에서 검색하였다. 결과 중 검색된 상위 100개를 크롤링하여 전문가 4명이 질의어에 맞는 정보인지 아닌지 판단하였다. 정보의 판별은 다수결에 의해 판별하였으며 만약 정보라고 판단한 수와 정보라고 판단하지 않은 수가 같을 경우에는 전문가의 연구 기간이 더 높은 사람에게 높은 비중을 할당하여 판단하였다. 이렇게 판단된 정보를 기반으로 해당 연구실 신입생의 질의어를 2010년 4월부터 2010년 8월 까지 약 5개월간 정보를 수집하였다. 이렇게 수집된 정보를 바탕으로 질의어 정보를 포함했을 때와 포함하지 않았을 때 검색을 비교하였다. 또한 1개의 단어를 이용하여 검색할 경우와 2개의 단어를 이용하여 검색할 경우 각각 적합한 페이지가 얼마나 증가하는지 비교 분석하였다.

5.1 질의어에 적합한 페이지 결정

페이지 랭크 알고리즘을 이용한 구글 검색엔진은 질의어와 웹 사이트의 구조 즉 인링크와 아웃링크를 이용하여 정보의 중요도를 측정한다. 하지만 웹 사이트의 구조적 특징 때문에 검색 질의어에 대한 부적합한 자료를 사용자에게 다수 추천해 준다. 예를 들어 "learning"이라는 단어로 질의했을 때 기업 홈페이지(http://www.bbc.co.uk/learning/)나 정부 홈페이지(http://lcweb2.loc.gov/learn/)와 같은 검색 질의어와 관련되지 않은 정보가 결과 순위 상위에 개재되어있다. 기업 홈페이지나 정부 홈페이지 같은 페이지들은 인링크와 아웃링크의 수가 다른 자료들 보다 많기 때문에 결과 순위 상위에 개재된다. 우리는 이러한 자료가 상위에 개제되는 것을 줄이기 위하여 주어진 질의어에 대해 워드넷 상에 상에 매칭되는 이전 질의어를 하위 집합으로 구성하여 이를 새로운 질의어에 사용하였다.

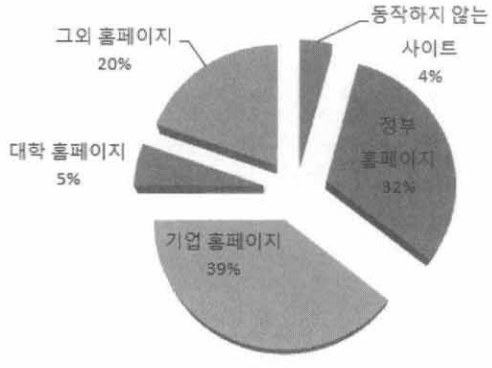
5.2 1개의 단어에 대한 적합한 페이지 비교

1개의 단어로 구성된 질의 15개를 구글 검색 엔진을 이용하여 상위 100개의 검색 결과를 크롤링하였다. <표 1>은 15개의 단어를 이용하여 구글에서 검색했을 때 사용자가 찾은 정보인지 아닌지 구별해 놓은 표이다. 표에서 나타나는 것처럼 15개의 단어로 검색된 결과 1500개 중 사용자에게 필요한 정보는 전체 문서의 40.4%인 606개의 문서이다. 그 이외의 문서들은 동작하지 않는 웹 페이지가 2.3%이고, 정부 홈페이지가 16.26%, 기업 홈페이지가 25.73%, 대학 홈페이지가 3.06%이며, 그 이외의 정보가 아닌 모든 페이지를 기타라고 하며 그 비율은 12.2%였다. (그림 6)은 <표 1>에서 나타낸 정보, 동작하지 않는 웹 사이트, 기업, 대학, 그 외의 비율을 그래프로 표현한 정보이다.

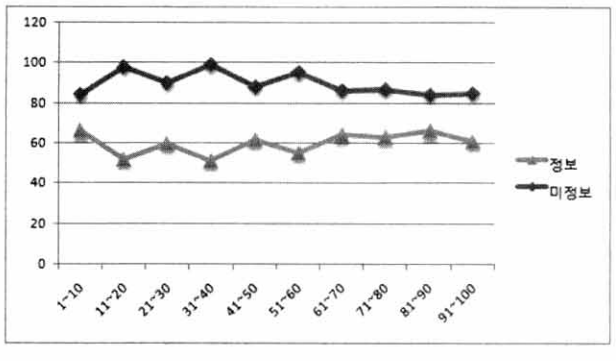
(그림 7)은 1500개의 문서 중 정보로 표현된 문서 606개가 어느 순위에 분포되었는지 알아보기 위해 100개의 순위를 10으로 나누어 해당 순위에 얼마나 많은 정보를 포함하는 지 나타내는 그래프이다.

〈표 1〉 1개의 단어를 이용한 적합한 페이지와 부적합 페이지에 대한 정보

검색 키워드	정 보	미 정 보 (홈페이지)					합계
		동작하지 않는 웹 사이트	정부	기업	대학	기타	
learning	16	3	28	30	1	22	100
machine	31	5	10	33	3	18	100
operation	27	0	50	21	0	2	100
individual	41	1	32	17	0	9	100
method	45	2	2	38	0	13	100
neural	52	5	8	21	8	6	100
reasoning	54	4	10	14	14	4	100
representation	62	2	9	8	1	18	100
system	27	1	32	34	5	1	100
conveyance	46	2	8	19	1	24	100
decision	46	1	10	33	1	9	100
graph	50	0	8	25	1	16	100
network	21	0	19	46	3	11	100
artificial	50	4	7	17	6	16	100
intelligence	38	5	11	30	2	14	100
total	606	35	244	386	46	183	1500



(그림 6) 15개의 단어로 검색된 문서의 유형

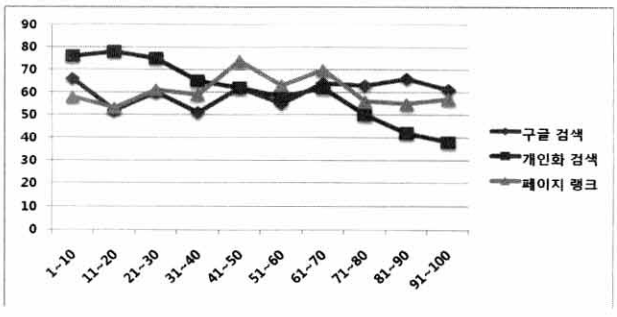


(그림 7) 15개의 단어로 검색된 1500개의 문서 분포

이러한 정보를 바탕으로 사용자의 새로운 질의어와 워드넷의 하위 집합에 포함된 이전 질의어를 찾아내서 질의어에 포함 시켜 벡터 모델 값을 구하였다. 또한 페이지 랭크 지수를 구하여 두 개의 값을 계산하여 검색 결과의 순위를 재조정하였다.

(그림 8)은 제안한 방법과 페이지 랭크 지수만을 고려한 경우, 구글 검색에서 검색된 결과를 비교 평가한 결과이다. 여기서 x축은 문서의 순위이고, y축은 15개의 질의어가 나타내는 정보의 수를 나타낸다.

상위 30개에 표현된 정보의 수를 살펴보면 제안한 방법이 229개의 정보를 표현해 주고, 구글이 178개의 정보를 표현해 주며 페이지 랭크 방법만을 이용했을 때 174개의 정보를 표현해 주고 있다. 상위 30개에 대한 정확률을 살펴보면 페이지 랭크가 38.2%이며 구글 검색이 39.5%, 제안한 방법이 50.8%로 위의 두 가지 보다 성능이 뛰어나다. 또한 (그림 8)에서 나타나는 것처럼 상위 40개에 나타나는 정보가 제안한 방법이 더 많음을 볼 수 있다.



(그림 8) 15개의 질의어에 대한 비교 평가

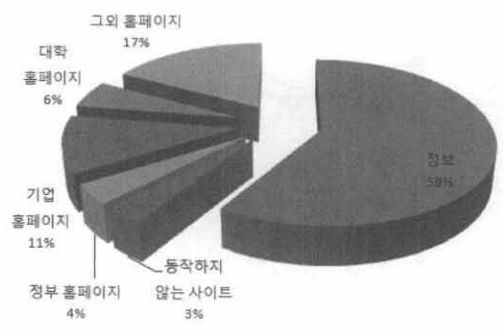
5.3 2개의 단어에 대한 적합한 페이지 비교

1개의 단어들을 2개의 단어로 묶어 정보의 양이 어떻게

〈표 2〉 2개의 단어를 혼합한 질의어 10개에 대한 적합한 페이지와 부적합한 페이지에 대한 정보

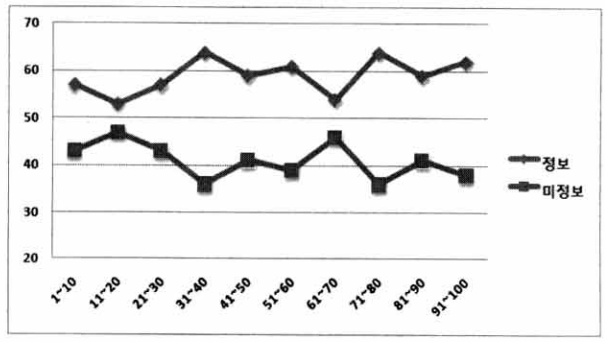
(2개 단어)혼합질의어	정 보	미 정 보 (홈페이지)					합계
		동작하지 않는 웹 사이트	정부	기업	대학	기타	
machine learning	48	7	6	5	14	20	100
operating system	53	1	12	20	7	7	100
conveyance individual	47	0	1	3	0	49	100
neural method	66	4	0	2	0	28	100
reasoning representation	65	3	3	1	10	18	100
decision graph	70	3	2	14	2	9	100
artificial intelligence	35	5	6	24	18	12	100
neural network	56	2	6	17	2	17	100
intelligent network	60	4	1	25	1	9	100
representation method	90	3	2	1	0	4	100
total	590	32	39	112	54	173	1000

변하는지 살펴보고 제안한 방법, 페이지 랭크, 구글 검색과 비교 평가 하였다. 2개의 단어로 구성된 혼합 질의어로 구글에서 검색하여 상위 100개의 검색 결과를 크롤링하였다. <표 2>는 2개의 단어로 구성된 혼합 질의어를 이용해서 구글에서 10번 검색했을 때 사용자가 찾는 정보인지 아닌지 구별해 놓은 표이다. 표에서 나타나는 것처럼 10번 검색시 검색된 결과 1000개 중 사용자에게 필요한 정보는 전체 문서의 59%인 590개의 문서이다. 그 이외의 문서들은 동작하지 않는 웹 페이지가 3.2%이고 정부 홈페이지가 3.9%, 기업 홈페이지가 11.2%, 대학 홈페이지가 5.4%이며, 그 이외의 정보가 아닌 모든 페이지를 기타로 하여 그 비율은 17.3%였다. 결과적으로 1개의 단어로 검색한 경우보다 2개의 단어로 검색한 경우 정보의 비율이 더 높음을 알 수 있다. 또한 이전의 비율이 높았던 정부 홈페이지나 기업 홈페이지의 수는 줄어들고 기타의 경우가 증가함을 볼 수 있다. 이는 질의어의 수가 많아질수록 홈페이지의 비율이 줄어든다는 것을 의미한다. (그림 9)는 <표 2>에서 나타난 정보, 동작하지 않는 웹 사이트, 정부, 기업, 대학, 기타 비율을 그래프로 표현한 것이다.



(그림 9) 10개의 혼합질의어로 검색된 문서의 유형

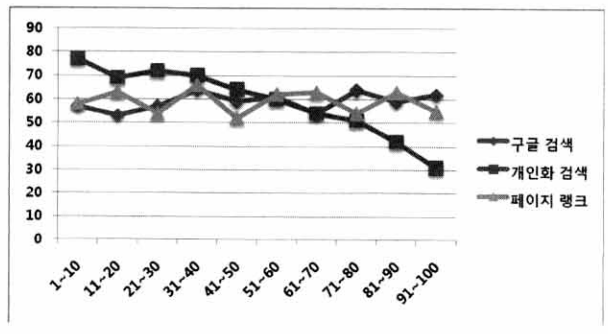
(그림 10)은 정보로 표현된 문서 590개가 어느 순위에 분포되었는지 보기 위해 100개의 순위를 10 단위로 구분하여



(그림 10) 10개의 혼합질의어로 검색된 1000개의 문서

해당 순위 중에 얼마나 많은 정보를 포함하는지 나타내는 그래프이다.

이러한 정보를 바탕으로 사용자의 새로운 질의어와 워드넷의 하위집합에 포함된 이전 질의어를 찾아내서 질의어에 포함 시켜 벡터 모델 값을 구하였다. 또한 페이지 랭크 지수를 구하여 두 개의 값을 계산하여 검색 결과의 순위를 재조정하였다. (그림 11)은 제안한 방법과 페이지 랭크 지수만을 고려한 경우, 구글 검색에서 검색된 결과를 비교 평가한 결과이다. 여기서 x축은 문서의 순위이고 y축은 10개의 혼합질의어가 나타내는 정보의 수를 나타낸다.



(그림 11) 혼합질의어 10개에 대한 비교 평가

상위 30개에 표현된 정보의 수를 살펴보면 제안한 방법이 218개의 정보를 표현해 주고, 구글이 167개의 정보를 표현해 주며 페이지 랭크 방법만을 이용했을 때 175개의 정보를 표현해 주고 있다. 상위 30개에 대한 정확률을 살펴보면 페이지 랭크가 58.3%, 구글 검색이 55.7%, 제안한 방법이 72.6%로 위의 두 가지 보다 성능이 뛰어나다. 또한 (그림 11)에서 나타나는 것처럼 상위 순위에 다른 두 가지 방법보다 더 많은 정보를 제공해 주고 있음을 알 수 있다.

6. 결론 및 향후 과제

6.1 결론

인터넷 정보 검색에서 중요한 문제는 검색 결과의 질과 관련된 것으로 검색 결과에 순위를 정하는 문제이다. 일반적으로 검색을 수행하여 검색 결과로 나오는 문서조차 도저히 다 읽어 볼 수 없을 정도로 많기 때문이다. 사용자들이 검색 엔진을 사용하는 경향을 살펴보면 모든 검색 결과를 참조하는 것이 아니고 상위에 개제된 몇 개의 문서만을 보고 만족하지 못하면 재검색을 하는 경향이 있다. 따라서 검색 엔진의 성능을 평가하는 주요 척도는 검색 결과의 상위 부분에 사용자의 요구와 일치하는 문서의 수에 초점이 맞춰져야 한다. 재현률(Recall) 보다는 정확률(Precision)이 검색 엔진의 성능에 더 중요한 요소이다. 이를 근거로 본 논문에서의 실험은 상위 30개의 검색 순위에서 정보가 나타나는 빈도를 측정하여 페이지 랭크와 구글에서 제공하는 검색 엔진 그리고 본 논문에서 제안하는 방법을 비교하였다. 1개의 단어인 경우 페이지 랭크보다 12.6% 정도 더 좋은 성능을 나타냈으며 구글 검색 보다는 11.3% 더 좋은 성능을 나타내었다. 의미가 좀 더 명확한 2개의 혼합질의어에서는 페이지 랭크보다 14.3%정도 더 좋은 성능을 나타내었으며 구글 검색 보다는 16.9%정도 더 좋은 성능을 나타내었다. 이것은 기업 홈페이지나 정부 홈페이지 같은 인링크와 아웃링크의 수가 일반 자료들 보다 많기 때문에 결과 순위 상위에 개재되었는데 제안된 방법에 의해 검색 하위에 개재되어 검색 성능이 향상된 것을 의미한다.

6.2 한계점 및 향후 과제

구글 검색 엔진과 비교 실험을 하였기 때문에 페이지 랭크 지수 또한 구글에서 제공하는 수치를 사용할 수밖에 없었다. 또한 여기서 제공하는 페이지 랭크 지수는 소수점 1의 자리까지 밖에 제공하지 않아 좀 더 세밀한 실험을 할 수 없었다. 향후 링크 정보를 가지는 문서 집합을 만들어 대중적 가치가 가지는 의미를 세밀하게 실험하여 평가하고자 한다.

참 고 문 헌

[1] SouMen Charkrabati., "mining the web Discovering Knowledge from Hypertext Data", Morgan Kaufmann Publishers. 2003.

[2] B. J. Jansen., A. Spink., T. Saracevic., "Real life, real users, and real needs: astudy and analysis of user queries on the web." Information Processing and Management. Vol.36, pp.207-227, 2000

[3] A. N. Langville., C. D. Meyer., "Google's PageRank and Beyond : The Science of Search Engine Rankings". Princeton University Press, 2006.

[4] F. Tanudjaja., L. Mui., "Persona: A Contextualized and Personalized Web Search." Proc. Of Int. Conf. on System Sciences, Vol.3, pp.53-61, 2002.

[5] Y. Sun., H. Li., I. G. Councill., J. Huang., "Personalized Ranking for Digital Libraries Based on Log Analysis." WIDM'08, pp.133-140, 2008.

[6] Z. Zhuang., S. Cucerzan., "Re-ranking search results using query logs", CIKM'06, pp.860-861, 2006

[7] U. Rohini., V. Ambati., "A collaborative filtering based re-ranking strategy for search in digital libraries", Lecture notes in computer science, pp.194-203, 2005.

[8] T. H. Haveliwala., "Topic-Sensitive PageRank: A Context-Sensitive Ranking Algorithm for Web Search", IEEE Transaction on Knowledge and Data Engineering, Vol. 15, No.4, pp.784-796, 2003.

[9] D. Fogaras., B. Racz., K. Csalogany., T. Sarlos., "Towards Scaling Fully Personalized PageRank: Algorithms, Lower Bounds and Experiments," Internet Math., Vol.2, No.3, pp.333-358, 2005.

[10] F. Qiu, J. Cho., "Automatic Identification of User Interest For Personalized Search", WWW 2006, pp.22-26, May, 2006.

[11] http://www.dmoz.org

[12] Mingjun. Lan., Shui. Yu., Ruth. Backer., Walei. Zhou., "A Co-Recommendation Algorithm for Web Searching.", Fifth International Conference on Algorithms and Architectures for Parallel Processing(ICA3PP'02). IEEE International Conference. 2002

김 태 환

e-mail : kimth@hanyang.ac.kr

2005년 인천대학교 컴퓨터공학(학사)

2007년 한양대학교 컴퓨터공학(석사)

2007년~현 재 한양대학교 컴퓨터공학 박사과정

관심분야 : 웹마이닝, 웹지능, 정보추출, 정보검색, 시맨틱웹과 온톨로지, 인공지능

전 호 철

e-mail : hochuls@chollian.net
1998년 서원대학교 전산계산학(학사)
2000년 한양대학교 컴퓨터공학(석사)
2010년 한양대학교 컴퓨터공학(박사)
관심분야 : 지능형 에이전트, 정보검색/정보추출, 인공지능, 상황인지

최 중 민

e-mail : jmchoi@hanyang.ac.kr
1984년 서울대학교 컴퓨터공학(학사)
1986년 서울대학교 컴퓨터공학(석사)
1993년 State University of New York at Buffalo, Computer Science(박사)
1993년~1995년 한국전자통신연구원(ETRI) 인공지능연구실 선임연구원
1995년~현 재 한양대학교 컴퓨터공학과 교수
관심분야 : 웹 마이닝, 웹지능, 정보추출, 시맨틱웹과 온톨로지, 인공지능