# 다중색인에 의한 정보검색 시스템 구현

이준영, 강상배, 양장모, 박 승, 박현주, 김민정, 권혁철
부산대학교 전자계산학과

# Implementation of an Information Retrieval System with Multiple Indexing

Jun-Young Lee, Sang-Bae Kang, Jang-mo Yang, Seung Park, Hyun-Joo Park,
Min-Jung Kim, Hyuk-Chul Kwon
Dept. of Computer Science, Pusan National University

요약

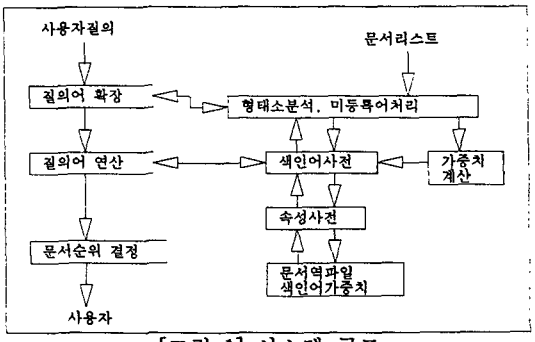
이 논문에서는 대량의 신문기사나 일반 텍스트 문서를 효율적으로 저장 및 검색 할 수 있는 정보검색 시스템을 구현한다. 이 시스템은 문서의 주제, 저자, 날짜, 출판사 또는 사용자 정의에 의한 속성과 본문에 대한 색인어와 색인관련정보를 생성한다. 모든 색인어는 최대 64가지의 속성정보와 문서별 단어빈도(tf)를 가질 수 있다. 색인은 형태소 분석을 이용하는 방법과 N-gram을 이용하는 방법이 동시에 사용되며, 색인어는 가중치를 가진다. 이 논문에서 구현한 시스템을 이용하여 7개월치 신문자료를 색인한 결과, 생성된 데이터베이스의 크기는 원래 문서의 약 22%이며 문서의 개수가 증가함에 따라 점점 그 비율은 감소한다.

## 1. 서론

현대 사회의 다양한 종류의 정보와 방대한 양의 정보를 사용자의 요구에 따라 정확하고 빠르게 찾아줄 수 있는 정보검색 시스템의 개발은 필수적이다. 신문기사나 일반잡지, 인터넷 상의 홈페이지, 특정 전문분야의 논문 등 다양한 문서집단의 데이터베이스를 구축하여 사용자가 관심 분야의 자료를 보다 융통성 있게 검색할 수 있는 검색 시스템이 요구된다. 또한 방대한 양의 자료를 처리함에 따르는 저장공간의 최소화와 그에 따른 검색속도를 증가시킬 수 있어야 한다.

## 2. 시스템 구조

이 논문에서 구현한 시스템은 색인기, 질의처리기, 파일처리의 세 부분으로 크게 나누어진다[그림 1]. 색인기는 형태소분석기[3]와 미등록어처리기[4]로 구성되어 있다. 질의처리기에서는 질의문 확장, 백터모델에 의한 검색, 불리언 연산, 검색된 문서에 대한 순위 결정을 한다. 파일처리부분은 색인어, 속성, 문서역파일, 색인어 가중치 등의 삽입, 삭제, 탐색을 제공한다.




[그림 1] 시스템 구조

각 문서는 최대 64개의 속성을 가질 수 있다. 속성은 사용자가 원하는 방법으로 문서를 검색할 수 있게 한다. 제목, 날짜, 저자, 사용자 정의 속성 등 문서의 속성을 본문에서 추출된 일반 색인어와 같이 처리함으로써 부가되는 저장공간을 최소화하였다.

* 본 연구는 '93년도 제3차 한국 과학 재단 목적 기초 협력 연구과제 연구비에 의해 연구되었음.

## 3. 색인

### 3.1 다중 색인을 이용한 색인어 추출

일반화된 색인 방법은 문서 집단의 특성을 반영하기가 어렵고, 사용자의 요구를 적절히 조절할 수가 없다. 따라서 색인방법을 다양화함으로써 사용여건과 문서 집단에 따라 융통성있는 색인 방법을 선택할 수 있어야 한다.

기존 색인방법이 사용하는 형태소 분석은 형태소 분석 규칙과 형태소 분석 사전을 이용해서 어절의 형태소를 분석하고, 분석된 결과를 바탕으로 색인어를 추출한다. 그러나, 형태소 분석은 어절내의 정보만 이용하므로 과분석되거나 모호성이 발생하여 지나치게 많은 분석 결과를 얻을 가능성이 있다. 따라서 앞 뒤 어절의 분석결과를 이용하여 과분석이나 모호성을 제거하고 올바른 색인어를 추출해야 한다. 또한 고유명사는 특성상 단어의 수가 많고 계속 만들어지며 응용하는 분야나 지역에 따라 그 활용이 현저하게 차이가 나므로 고유명사를 모두 사전에 등록하는 것은 불가능하다. 한편, 형태소 분석기는 사전 정보를 바탕으로 문장을 분석하므로 사전에 등록되지 않은 미등록어를 포함한 어절은 분석이 되지 않는다. 이런 단점을 해결하기 위하여 추정을 통해 미등록어를 색인어로 추출해야 한다. 그러나 추정된 색인어는 형태소 분석에 의한 색인어에 비해 검증되어지지 않은 명사이므로 잘못 추정될 가능성이 존재한다. 따라서, 이런 추정의 한계를 극복할 수 있는 방법으로 미등록어가 포함된 어절에서 최장의 조사를 뗀 명사에 대해서 다이어그램(diagram)을 추출하는 방법이 있다.

샘플 데이터(부산일보 신문기사 중에서)

| |
|---|
| 생곡쓰레기매립장 부근의 주민들은 부산시의 쓰레기반입에 대한 반대시위를 ... |

#### ① 형태소 분석을 통한 명사의 추출.

형태소 분석 사전과 규칙을 기반으로 하여 어절의 형태소를 분석한다. 어절의 중의성에 의해 다수개의 결과로 분석되는데, 이 결과들 중에서 명사가 포함된 결과에서 명사를 추출하여 색인어로 선택한다. 형태소 분석 사전에 의해 검증된 명사를 색인어로 추출하는 장점이 있다. 위에서 주어진 문장을 색인한 결과는 다음과 같다.

| | |
|---|---|
| 색인어 | 부근, 주민, 부산시, 쓰레기, 쓰레기반입, 반입, 대한, 반대, 대한반대, 반대시위, 시위, ... |

#### ② 형태소 분석과 중의성 제거를 통한 명사의 추출.

형태소 분석에서 발생하는 중의성을 제거하여 보다 정확한 색인어를 추출한다. 형태소 분석의 결과들 중에서 가장 적절한 결과를 선택하는 과정을 중의성 제거 과정이라고 하며, 이 선택된 결과에 포함된 명사를 색인어로 추출한다. 중의성 제거 방법에 의한 속도 저하에 비해, 가장 적절한 결과에서 명사를 선택하므로 보다 정확한 색인어를 추출할 수 있다.

| | |
|---|---|
| 색인어 | 부근, 주민, 부산시, 쓰레기, 쓰레기반입, 반입, 반대, 반대시위, 시위, ... |

#### ③ 형태소 분석, 중의성 제거, 그리고 미등록어 추정을 통한 명사의 추출.

①과 ②의 방법에서 추출할 수 없는 미등록어는 추정을 통하여 색인어를 선택한다. 고유명사가 많이 포함된 문서 집단들, 예를 들어 신문기사, 소설, 전공문서 등의 문서집단에서는 형태소 분석이나 중의성 제거의 방법으로는 선택할 수 없는 명사를 미등록어 추정을 통하여 색인어를 추출할 수 있다. 대신 추정을 통한 미등록어를 색인어로 추출하므로 미등록어의 경우 색인어의 정확성이 떨어지는 단점이 있다.

| | |
|---|---|
| 색인어 | 생곡쓰레기매립장, 부근, 주민, 부산시, 쓰레기, 쓰레기반입, 반입, 반대, 반대시위, 시위, ... |

#### ④ 형태소 분석, 중의성 제거, 미등록어 추정을 통한 명사의 추출과 미등록어가 포함된 어절에 대해선 최단 조사 절단법을 이용하여 추출된 명사를 다이어그램(diagram)화하여 명사를 추출.

형태소 분석, 중의성 제거, 미등록어 추정을 통한 명사 추출에 더하여, 미등록어를 포함하는 어절

에서 최장의 조사와 서술격 조사와 어미의 결합꼴을 제거한 후 남은 스트링(string)을 다이어그램(diagram)으로 나누어서 그 각각의 다이어그램(diagram)을 색인어로 선택한다. 예를 들어 '프로그래밍'을 다이어그램(diagram)화하면 '프로', '로그', '그래', '래밍'으로 구성된다.

| | |
|---|---|
| 색인어 | 생곡쓰레기매립장, 부근, 주민, 부산시, 쓰레기, 쓰레기반입, 반입, 반대, 반대시위, 시위, ... |
| N-gram | 생곡, 곡쓰, 쓰레, 레기, 기매, 매립, 립장 |

다이어그램(diagram)을 이용하면 색인어의 개수가 증가하고, 관련성이 없는 문서도 검색하는 단점을 가지고 있다. 반면, 미등록어를 포함하는 어절에서 다이어그램을 추출하므로, 미등록어 추정기에서 추정하지 못한 어절에서 색인어를 추출하므로 보다 나은 검색효율을 얻을 수 있다.

### ⑤ 최장 조사 절단법을 이용하여 얻어진 명사를 다이어그램(diagram)화하여 명사를 추출.

명사를 포함하는 어절에서 최장의 조사를 절단하고, 추출된 명사를 다이어그램(diagram)화하여 색인어를 추출한다. 이 방법은 다른 방법들에 비해 명사 사전, 조사 사전과 몇가지의 규칙만을 사용하므로 색인어를 추출 시간이 대단히 적게 걸리는 장점을 가지나, 형태소 분석기를 이용하는 다른 색인방법에 비해 적절하지 않은 색인어가 많이 추출되는 단점이 있다.

| | |
|---|---|
| N-gram | 생곡, 곡쓰, 쓰레, 레기, 기매, 매립, 립장, 부근, 주민, 부산, 산시, 기반, 반입, 대한, 반대, 시위, ... |

이 논문에서 구현한 시스템은 사용자가 위의 다섯가지 검색모드를 선택할 수 있게 하여 사용자 요구에 맞는 검색을 제공한다.

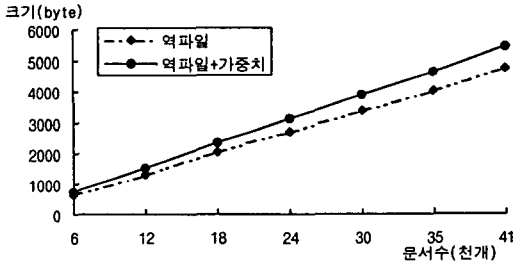
## 4. 정보의 저장

검색 시스템에서 정보의 저장구조는 검색속도 및 저장공간의 효율성을 크게 좌우한다. 일반적인 정보의 저장구조는 색인어와 이들 색인어를 포함하는 문서번호들로 구성된 역파일(inverted file)을 이용한다[1]. 이 시스템에서는 색인어와 각 색인어에 대한 속성정보, 그리고 각 색인어별 속성에 따른 문서 역파일로 구성된다.

문서역파일은 이진 역파일(binary inverted file)의 형태인 비트벡터(bit-vector)로 구성한 후, 이 비트벡터를 $1/\rho$에 의한 블럭크기를 가지는 상대거리를 이용한 1계층 비트벡터 압축 기법[참고논문]을 이용하여 압축저장한다. 이 방법은 기존의 run-length 압축기법[11]보다 13.65%, 기존의 prefix-omission을 이용한 1계층 비트벡터 압축기법[8]보다 1.88%의 압축효율이 뛰어난 기법이다[2]. 색인어 가중치를 같이 저장하여 최소한의 저장공간으로 불리언 검색 모델과 벡터 검색 모델을 모두 지원한다. 벡터모델의 검색을 지원하기 위해 이 시스템에서는 색인어 가중치로써 문서내에 나타난 색인어의 출현빈도(*tf*, term frequency)를 이용한다. 색인어 가중치는 이진역파일에 가중치를 주어 저장한다.

| 문서수 (개) | 색인어수 (개) | 문서역파일 크기(byte) | 빈도 정보 크기(byte) |
|---|---|---|---|
| 6,729 | 122,163 | 629,066 | 104,304 |
| 12,776 | 188,430 | 1,302,291 | 205,868 |
| 18,689 | 247,542 | 2,009,489 | 311,113 |
| 24,265 | 292,863 | 2,682,967 | 410,491 |
| 30,041 | 334,696 | 3,380,562 | 511,784 |
| 35,284 | 368,484 | 4,020,077 | 606,581 |
| 41,025 | 407,678 | 4,734,644 | 711,591 |

[표 2] 문서역파일과 가중치저장 크기 비교




[그래프 1] 색인어 가중치를 저장할 때 추가로 요구하는 저장공간

[표 2]는 각 색인어의 문서역파일과 색인어의 저장공간의 크기를 나타낸 것이다. 생성된 데이터베이스의 크기비교를 위해 색인은 형태소분석을 통해 명사만을 추출한 방법을 적용했다. 색인어 가중치로써 빈도정보를 저장할 때 추가로 요구되는 저장공간은 문서 역파일에 비해 약 15%이다[그래프 1].

## 5. 검색

이 논문에서 구현한 검색시스템은 불리언 질의와 자연어 질의를 모두 제공한다. 자연어 질의는 색인어를 추출하는 방법에 따라 질의어를 추출하고, 추출된 질의어는 OR연산을 사용하여 불리언 질의문을 구성한다. 불리언 질의문에서는 복합명사가 포함될 경우에 복합명사를 단일명사로 분리하고, 분리된 단일명사들은 OR연산을 사용하여 질의문을 재구성한다. 연산된 결과에 대해서 코사인 유사계수 함수를 이용하여 문서의 순위(ranking)을 결정한다.

색인어의 가중치는 단어빈도($tf_{ik}$, term frequency)를 사용한다. 질의어의 가중치는 문서집단내의 문헌빈도($df$ : document frequency)의 역의 값을 사용한다. 질의문 $j$에 포함된 질의어 $k$의 가중치($q_{jk}$)는 아래와 같다.

$$q_{jk} = \frac{1}{df_{jk}}$$

$df_{jk}$는 질의어 $k$의 문서집단내에서의 문헌빈도이다. 이 질의어 $k$가 문서집단내에 포함되지 않으면, 질의어 $k$의 가중치 $q_{jk}$는 0의 값을 가진다. 문서와 질의문의 유사도를 측정하기 위해 코사인 유사계수 함수를 이용한다[9]. 코사인 유사계수 함수는 다음과 같다. $t_{ik}$는 문서 $i$에 포함된 색인어 $k$의 가중치이고, $q_{jk}$는 질의문 $j$에 포함된 질의어 $k$의 가중치이다.

$$S(D_i, Q_j) = \frac{\sum_{k=0}^{t} t_{ik} \cdot q_{jk}}{\sqrt{\sum_{k=0}^{t} (t_{ik})^2 \cdot \sum_{k=0}^{t} (q_{jk})^2}}$$

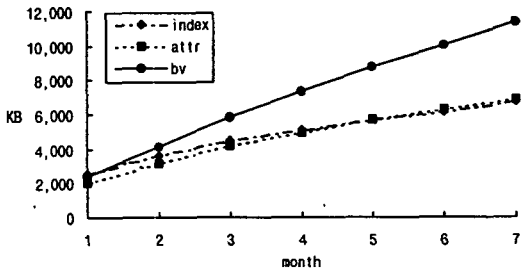
$$D_i = (t_{i1}, t_{i2}, \cdots, t_{it})$$
$$Q_j = (q_{j1}, q_{j2}, \cdots, q_{it})$$

## 6. 실험 및 결과분석

이 논문에서 제시한 다중색인 모드에서 형태소 분석을 통해 명사만 추출한 방법으로 신문기사 41,025개에 대해 데이터베이스를 구축하여 분석한 것이다. 문서내에 최대 64개의 속성을 정의하여 사용하고 있으며 각 추출된 색인어는 1개 이상의 속성값을 가진다. [표 3]은 구축된 데이터베이스의 크기를 신문기사 파일과 비교한 것이다.

| 문서수 (개) | 색인어 사전(KB) | 속성사전 (KB) | 역파일 (KB) | 문서파일 (KB) |
|---|---|---|---|---|
| 6,729 | 2,559 | 2,060 | 2,397 | 12,563 |
| 12,776 | 3,594 | 3,184 | 4,159 | 24,349 |
| 18,689 | 4,470 | 4,190 | 5,841 | 36,552 |
| 24,265 | 5,119 | 4,963 | 7,320 | 47,728 |
| 30,041 | 5,713 | 5,681 | 8,750 | 59,114 |
| 35,284 | 6,175 | 6,263 | 10,013 | 70,380 |
| 41,025 | 6,733 | 6,933 | 11,414 | 82,366 |

[표 3] 구축된 데이터베이스

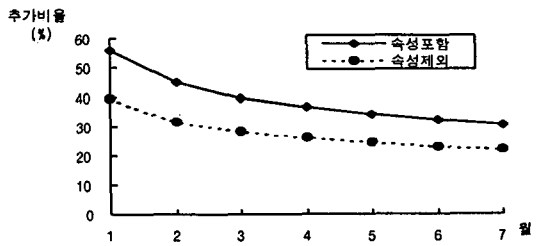



[그래프 2] 사전 크기의 증가율.

문서의 개수가 증가함에 따라 추출된 색인어 개수의 증가율은 점점 작아진다[그래프 2]. 그에 따른 색인어 사전과 속성사전 크기의 증가율도 점점 줄어든다[그래프 2]. 문서역파일의 크기는 문서 개수의 증가에 따라 점점 증가한다[그래프 2].

| 문서수(개) | 6729 | 12776 | 18689 | 24265 | 30041 | 35284 | 41025 |
|---|---|---|---|---|---|---|---|
| 속성포함(%) | 55.8 | 44.9 | 39.7 | 36.5 | 34.1 | 31.9 | 30.4 |
| 속성제외(%) | 39.5 | 31.8 | 28.2 | 26.1 | 24.5 | 23.0 | 22.0 |

[표 4] 문서수의 증가에 따른 추가메모리의 증가율




[그래프 3] 추가 메모리의 증가율 그래프

[표 3]에서 문서역파일의 구조는 이 논문에서 제안한 색인어의 빈도(tf)를 압축하는 방법으로 같이 저장되어 있다. 따라서 벡터모델에서 만들어지는 데이터베이스의 크기는 원시 문서 파일에서 약 30%의 추가 메모리를 요구하며[표 4], 문서 개수가 증가함에 따라 그 비율은 점점 작아진다[그래프

3]. 이 시스템에서 속성을 가지지 않고 단순히 본문색인을 통해서 데이터베이스를 구축한다면 약 22%의 추가 메모리를 필요로 한다.

## 7. 결론

이 논문에서 구현한 시스템은 다중모드 색인과 속성에 따른 색인어를 제공함으로써 문서집단의 특성과 사용자의 요구 방법에 따른 검색이 가능하다. 불리언 모델을 확장하여 벡터모델을 제공하기 위해 색인어 가중치를 저장했으나, 단지 추가의 15%의 저장공간만이 필요했다. 이 논문에서 구현한 시스템에서 생성되는 사전의 크기는 문서파일의 약 22%이며 문서개수가 증가함에 따라 그 비율은 점점 줄어들었다.

앞으로 이 시스템의 검색 효율에 대한 성능 평가가 필요하다. 이 논문에서 구현한 다중색인에 의한 검색은 신문기사의 경우 미등록어가 많이 포함되어 있으므로 효율적일 것으로 예상된다. 신문기사에 대한 성능 평가를 위해서는 전문가가 모든 기사를 수작업으로 관련 문서를 판단해야 하는 어려움이 있다.

## 8. 참고문헌


[1] 정영미, *정보검색론*, 정음사, 1988.

[2] 이준영, 권혁철, "*문서검색 시스템을 위한 도치색인파일의 압축저장기법 개선*", 한글 및 한국어 정보처리 학술발표 논문집, pp. 18-22, 1995.

[3] 김민정, 권혁철, "*한국어 특성을 이용한 자동색인 기법*", 정보과학회 가을 학술 발표 논문집, pp. 1005-1008, 1992.

[4] 양장모, 권혁철, "*언어정보를 이용한 한국어 미등록어 추정*", 정보과학회 봄 학술발표 논문집, pp. 957-960, 1996.

[5] 김문석, 신동욱, "*복합명사 통계자료를 이용한 한글 자동색인 시스템 개발*", 정보과학회 봄 학술발표 논문집, pp.931-934, 1996.

[6] Min-Jung Kim, Hyuk-Chul Kwon, "*One Level Bit-Vector Compression with Relative Addressing*", Information Retrieval with Oriental Languages, pp. 96-101, 1996.

[7] William B. Frakes, Ricardo Baeza-Yates, *Information Retrieval Data Structures & Algorithms*, Prentice Hall, 1992.

[8] Y. Choueka, A.S. Fraenkel, S.T. Klein & E. Segal (1986), "*Improved Hierarchical Bit-Vector Compression in Document Retrieval Systems*," Organization of the 1986-ACM Conference on Research and Development in Information Retrieval, 88-96, 1986.

[9] Gerard Salton and Michael J. McGill, *Introduction to Modern Information Retrieval*, NewYork: Mcgraw-Hill, 1983.

[10] Alran L. Thrap(1988), *File Organization And Processing*, John Wiley & Sons, Inc, 1988.

[11] Alistair Moffat & Justin Zobel (1992). "*Parameterised Compression for Sparse Bitmaps*," In Proc. 15'th ACM-SIGIR Conference on Research and Development in Information Retrieval, 274-285, 1992.