

필드 구조 문서를 위한 교차 필드 검색 모델

윤보현, 왕지현, 강현규,
한국전자통신연구원 언어공학부

Cross Field Searching Model for Field Structured Documents

Bo-Hyun Yun, Ji-Hyun Wang, Hyun-Kyu Kang
Linguistic Engineering Department, ETRI

요약

기존의 전문 검색 시스템은 문서를 단지 단어의 연속이라는 제한적 관점에서만 바라보았다. 또한 기존의 필드 검색 시스템은 고정된 필드를 색인 및 검색대상으로 하거나, 문서의 내용이 아닌 메타 정보에 관한 검색만이 가능하였다.

본 논문에서는 내용과 필드 구조를 통합하여 가변 필드 구조 문서를 색인 및 검색하는 모델인 교차 필드 검색 모델을 제안한다. 기존 정보검색 시스템의 기능을 기본으로 제공하면서 필드구조를 색인/검색하기 위한 기능적 요구사항을 제시하고, 내용 및 필드 구조를 색인하면서 동적인 삽입/삭제가 가능한 색인 구조를 제안한다. 아울러 검색시에 문서 가중치를 계산하여 문서를 순위조정하는 불리언 모델, 확장 불리언 모델, 벡터 공간 모델의 변형 모델을 제시한다. 아울러 구현 사례로 STEER-XDS 검색 시스템에 대해 알아본다.

1. 서론

정보검색은 대량의 정보 속에서 정보사용자가 원하는 정보를 빠르게 찾아낼 수 있도록 도와주는 중요한 정보인프라 중의 하나이다. 정보검색시스템의 성능은 사용자가 원하는 정보를 얼마만큼 효율적으로 찾아내느냐에 달려있다. 이러한 정보검색시스템의 성능은 두 가지의 구성요소를 갖고 있다. 정보를 얼마나 빠르게 찾아내느냐에 관한 정보탐색(searching) 요소와 더불어 정보검색시스템 자체의 정보표현 능력이 어느 정도인가에 관한 정보표현(representation or indexing) 요소가 있다.

최근 도서, 논문 또는 특허정보 등에 대한 전문(full-text)이 전자화되면서 상당한 분량의 전문 자체에서 특정부분만을 검색하고자 하는 사용자의 요구가 증가되었다. 예를 들어, 그림 1과 같은 여러 개의 가변 필드 구조를 가지는 형태의 문서를 색인 및 검색하는 검색 시스템을 고려해 보자. 아

무리 대량의 도서를 빠르게 찾아내더라도 문서전체를 대상으로만 찾아낼 수밖에 없다면 정작 사용자는 자신이 원하는 정보를 빠르게 찾아 낼 수 없고, 수많은 검색결과 리스트를 헤매다가 검색을 포기하게 될 것이다. 그러나 만약 저자정보, 출판사, 혹은 요약정보로도 검색이 가능하다면 사용자는 자신이 질의할 수 있는 영역자체가 확장되므로 좀더 정확한 정보를 빠르게 찾아낼 수가 있다.

이와 같이 정보가 여러 필드에 분산되어 있을 때 검색어와 필드이름을 조합하여 문서나 필드들을 검색하는 방법을 “필드검색(field searching)”이라 한다. 또한 특정 필드를 제한하여 검색하기 때문에 이를 “제한검색(constraining searching)”으로도 일컫는다. “교차필드 검색(cross field searching)”이란 동시에 두 개 이상의 필드를 검색할 수 있는 방법을 의미한다. 예를 들어 한글요약에 “정보”라는 단어와 제목에 “검색”이라는 단어가 있는 문서를 찾고 싶을 때 다음과 같은 질의 (“정

보 in 한글요약 or 검색 in 제목)”를 할 것이다. 교차 필드 검색은 웹이나 도서관 자료 검색 등에 강력한 도구이며, 보다 나은 검색을 가능하게 한다. 따라서 교차필드검색의 지원 여부도 검색엔진을 차별화하는 요소이기도 하다.

```

<Domain name>
정보과학논문
<File name>
kiss9607.txt
<Title>
형태와 컬러성분을 이용한 내용 기반의 ...
<Date>
199960701
<논문지정보>
정보처리논문지 제33권 4호 1996.7
<영어제목>
Contents Based Image Retrieval Database
System using Shape and Color Feature
<1장>
내용 기반 정보 검색이란 멀티미디어 ....
<2장>
형태와 컬러 성분 ....
<3장>
시스템 구성도는 다음과 .....
<4장>
제한한 시스템의 실험 결과는 .....
<End doc>

```

그림1. 전자화된 필드 구조 문서의 예

기존 전문검색 시스템은 문서의 내용을 분석하여 색인어를 추출하고, 사용자의 질의가 주어졌을 때, 질의에 사용된 단어와 색인어의 유사성을 계산하여 얻어진 결과를 제공한다. 이러한 시스템의 문제점은 문서를 단지 단어의 연속이라는 제한적 관점에서 보았고, 구조화되지 않은 문서의 집합에 대해서만 질의를 수행하였다. 즉, 문서를 단어의 연속이라는 제한적 관점에서 보았기 때문에 문서내의 구조는 무시되었다. 이와 같은 검색은 사용자가 검색하고자 하는 문서의 부분을 지정할 수 없으며, 항상 문서 전체에 대해 검색하므로 검색 시간이 오래 걸리는 문제가 있다. 따라서 기존의 전문 검색 시스템은 문서 전체에 대한 전문검색만을 지원하며, 문서의 구조를 적절히 활용하지 못하는 단점이 있다[4, 10, 15, 16, 18].

본 논문에서는 가변 필드 구조를 갖는 문서를 검색하기 위한 기능적 요구사항을 살펴보고, 내용과 필드 구조를 색인 및 검색하는 교차필드검색

모델을 제시한다. 아울러 교차 필드 검색의 응용분야 및 개발사례를 살펴본다.

2. 필드 검색의 유형

필드 검색의 유형은 다음과 같이 검색식에 직접 필드 제한자를 붙이는 경우, 검색식을 입력한 다음 검색대상 필드를 옵션으로 선택하는 경우, 그리고 필드별 검색식 입력상자에 검색식을 작성하는 경우 세가지로 나뉜다.

1) 검색식에 직접 필드 제한자를 붙이는 경우는 국외 웹 검색엔진으로 AltaVista[1], DejaNews[2], Infoseek[7], Lycos[8], Yahoo[12]가 해당되며, 국내 웹 검색 엔진으로 한글라이코스[19], 한글알타비스타[20], 한글야후[21]가 해당된다. 또한 국내 전자도서관중에 국회도서관 검색엔진[13]이 이 유형에 속한다.

예를 들어, AltaVista 검색엔진에서 질의 "(title:information and site:kr)"는 제목에 "information"이라는 단어가 나오는 .kr site의 웹문서를 보고자 할 때 사용한다.

이 유형에 속하는 검색 엔진은 교차 필드 검색을 지원하지 않는다. 하지만 고정된 필드를 가지는 문서만을 대상으로 하고 있다. 또한 알타비스타를 제외하고는 문서 내용에 대한 필드 검색을 지원하지 않는다. 즉, 문서의 내용이 아닌 메타정보에 대한 필드 검색을 지원하지 않는다.

2) 검색식을 입력한 다음 검색대상 필드를 옵션으로 선택하는 경우는 검색엔진 Digital LINNET[3], Galaxy[5], Hotbot[6], 그리고 Open Text Power Search[9]가 해당된다.

3) 필드별 검색식 입력상자에 검색식을 작성하는 경우는 검색엔진 REHABDATA[11]과 유럽 사회과학 정보 협의회 통합 검색(CESSDA)[17]이 해당한다.

대부분의 국내 도서관 검색엔진들은 2)번과 3)번 유형에 속하는 필드 검색을 수행한다. 이러한 유형의 필드 검색은 교차 필드 검색을 수행할 수 없으며, 고정된 필드를 가지는 문서만을 대상으로 하고 있다. 또한 문서 내용에 대한 필드 검색을 지원하지 않고 메타정보에 대한 필드 검색만을 지원한다.

3. 교차 필드 검색 모델

3.1 기능적 고려사항

1) 기존 정보검색 시스템의 기능 제공

교차 필드 검색 시스템이 가장 최소한으로 제공해야 할 기능은 당연히 기존의 정보검색시스템이 갖는 모든 기능을 기본적으로 제공해야 한다는 것이다. 이러한 기존 정보검색 시스템의 기본 기능은 자동색인기능, 빠른 탐색기능, 문서순위화 기능 등을 포괄한다.

2) 필드단위 색인 및 검색 기능

정보구조화를 통해 논리적으로 구분된 정보단위에 대한 개별 검색이 가능해야 한다. 이는 검색결과가 문서단위가 아닌 필드별도 될 수 있음을 의미한다. 책 한권이 구조화되어 책 단위의 검색은 물론 장(chapter), 절(section)단위로도 개별 순위화가 가능하여 사용자가 분할된 정보단위로 검색결과를 제공받을 수 있어야 한다.

3) 동적인 삽입 및 삭제 기능

일괄적으로 검색대상의 문서를 색인하는 기능뿐만아니라 원할 때는 언제든지 문서를 색인하고 삭제할 수 있는 동적 색인 구조를 지원해야한다. 따라서 기존의 색인 정보의 수정을 최소화하여 효율적인 색인 정보의 관리를 도모할 수 있어야 한다.

4) 질의 파싱 기능

교차필드검색의 질의는 키워드, 필드이름, 불리언 연산자가 함께 사용되므로 이들을 적절히 인식할 수 있는 질의 파싱 기능이 필요하다.

3.2 교차 필드 색인

1) 색인어 추출

주어진 문자열을 형태소 해석하여 색인어로 사용할 수 있는 단어들과 이들의 위치 정보(문장 번호, 문장내 어절 번호)를 추출한다. 영어 단어의 경우 스템밍(stemming)하고 설정에 따라 대문자를 소문자로 바꾼다. 한자인 경우 설정에 따라 한글로 변환한다.

2) 색인 정보 저장

색인 정보로 포스팅 정보와 문서정보가 저장된다. 포스팅 정보로는 색인어의 출현 문서빈도(document frequency), 위치정보, 문서번호, 문서내 색인어빈도(term frequency), 필드번호, 필드내 색인어빈도가 저장된다. 문서정보로는 문서이름, 제목, 날짜, 필드개수, 필드번호, 필드내용길이, 필드

내용이 저장된다.

3) 색인 구조

본 논문에서 제안하는 내용과 필드 구조를 색인하는 교차 필드 색인구조는 그림 2와 같다. 이 구조는 검색과 삭제 속도를 증시한 구조로서 검색 속도를 증가시키기 위해 색인어당 하나의 포스팅 레코드와 위치정보 레코드를 가진다. 아울러 삭제 속도를 증시하기 위해 문서당 포스팅 레코드 지정파일을 두어 문서에 속한 색인어를 바로 찾아 삭제 가능하였다. 그러나 삽입에 있어서는 문서당 포스팅 레코드나 위치정보 레코드를 저장하기 위한 오버헤드가 존재한다.

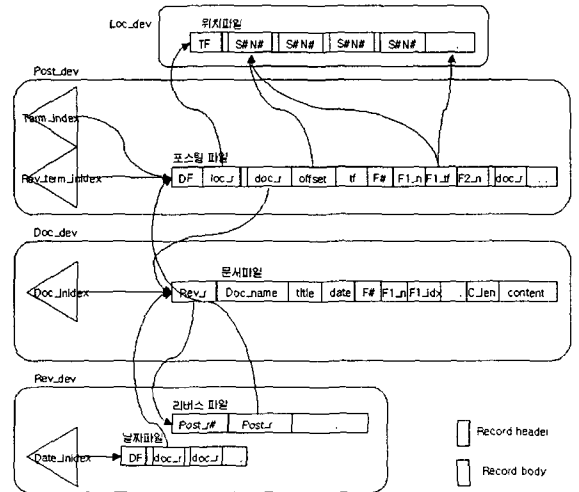


그림 2. 교차 필드 검색의 색인 구조

역색인을 구성하는 각 중요 구조들은 Loc_dev, Post_dev, Doc_dev, Rev_dev 4개의 분리된 device로 나누어진다. Term_index는 색인어와 포스팅 레코드의 B+ 트리 색인이며, Rev_term_index는 절단(truncation)처리를 위해 색인어를 리버스하는 색인이다. Doc_index는 문서의 이름과 내용 레코드를 가리키는 B+ 색인이며, Date_index는 날짜를 효율적으로 검색하기 위한 색인이다.

포스팅 파일은 각 색인어의 포스팅 정보를 저장하는 파일이며, 위치 파일은 검색 속도를 빠르게 하기 위해 각 색인어의 위치 정보를 저장하는 파일이다. 리버스 파일은 포스팅 레코드 개수와 실제 포스팅 레코드를 가리키는 정보를 저장하는 파일이다. 또한 문서 파일은 실제 문서의 내용을 저장

할 파일이며, 날짜 파일은 날짜-문서의 역색인 리스트를 갖는 파일이다.

3.3 교차 필드 검색

1) 교차 필드 검색 구조

검색 서버는 그림 3에서 보듯이 메모리 관리 모듈, 검색분배/통합 모듈, 세가지 검색 모듈로 이루어져 있다. 검색의 특성상 대량의 메모리를 잠깐 사용하고 다시 운영체제에 반환하는 작업이 반복적이고 운영체제에의 메모리 할당 요구가 시간 소모 작업을 고려하여 다사용자 접속시 검색 성능 저하 방지를 위한 메모리 관리모듈이 있다. 또한 검색한 중간 결과를 저장하고 분배/통합하는 모듈이 있으며, 실제 문서를 랭킹하기 위한 세가지 불리언 모듈, 확장 불리언 모듈, 벡터 공간 모듈이 있다.

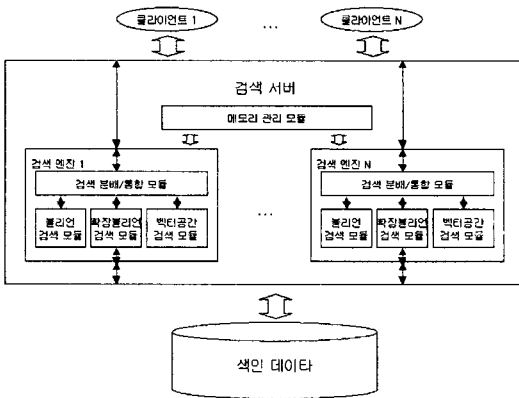


그림 3. 교차 필드 검색 구조

2) 질의 파싱

사용자의 질의를 받아서 질의 BNF 형태인 그림 4를 기반으로 Lex와 Yacc을 이용하여 step-query의 형태로 바꾼다. step-query란 사용자가 입력한 질의를 하나씩 분해하여 검색시스템이 사용할 수 있도록 하는 구조로 예를 들어 "AND 정보:0.7 in 한글요약 검색:0.5 in 제목"의 형태이다. 이것은 0.7의 가중치를 갖는 "정보"가 있는 한글요약과 0.5의 가중치를 갖는 "검색"이 있는 제목을 갖는 문서를 찾겠다는 의미이다.

복합명사의 경우는 이를 불리언 연산자를 사용하여 분리한 결과와 같이 생성하여 질의를 재구성한다. 예를 들어 "정보검색"이란 질의는 "(정보

AND 검색 OR 정보검색)"으로 재구성한 후 이를 step-query로 만든다. 영어는 스테머를 돌려서 질의를 만들며 대문자는 모두 소문자로 변환하는 역할도 수행한다.

그림 4에서 KEYWORD는 공백으로 구분되는 한 단어를 의미하며, WEIGHT는 10진수 digit 혹은 실수이다. 명사 tag는 nc(보통명사), nq(고유명사) 등이 사용된다. "AND, and, &"은 불리언 and를 수행하고, "OR, or, |"는 불리언 or를 의미하고, "ANDNOT, -"은 불리언 ANDNOT 수행한다. ":"는 질의어 가중치를 부여할 때 사용하고, "(,)"은 불리언 연산의 우선순위 표현하기 위해서 사용한다. "in"은 필드 검색을 수행하기 위한 필드 지정 연산자이고, "NEAR, near"은 "near term term number" 형식을 가지고 number내로 떨어진 두 단어를 검색하는 연산자이다. "WITHINS, within"은 "withins term term number" 형식으로 사용되어 number내 문장으로 떨어진 두 단어를 검색하는 연산자이다. "Date from to"는 date연산을 수행하기 위한 연산자이며 질의의 시작에만 가능하다. 아울러 질의어를 나열하면 벡터검색을 수행한다.

Query BNF form

```

Start      := date Vec_Query
Date       := date NUMBER NUMBER
Vec_Query  := Query Vec_query |
Query      := query AND query
            | query OR query
            | query ANDNOT query
            | sub_query_weight IN field
            | literal
            | literal in field
sub_query_weight := subquery : Weight
                | subquery
sub_query      := ( query )
literal        := term
                | near term term number
                | withins term term number
term           := KEYWORD
                | KEYWORD : WEIGHT
field          := KEYWORD
    
```

그림 4. 질의 BNF 형태

3) 문서 랭킹

질의어 qt_i 가 가중치 qw_i 를 갖는 질의 Q 는 다

음과 같이 표현된다.

$$Q = \{(qt_1, qw_1), \dots, (qt_i, qw_i), \dots, (qt_m, qw_m)\}$$

하나의 질의어 qt_i 에 대해 검색된 n 개의 결과인 문서집합 D 는 다음과 같이 표현된다.

$$D = \{(d_1, dw_1), \dots, (d_j, dw_j), \dots, (d_n, dw_n)\}$$

dw_j 는 문서 d_j 가 질의어 qt_i 에 대해 가지는 가중치를 의미한다.

질의어 qt_i 에 대한 문서 d_j 의 가중치 dw_j 는 다음과 같이 계산된다.

$$dw_j = qw_i \times \left(\frac{tf_j}{maxtf} \times \frac{1}{df_j} \right)$$

tf_j 는 질의어 qt_i 가 문서내에서 나타난 색인어 빈도이고, df_j 는 질의어 qt_i 가 전체 문서에서 나타난 문서 빈도이며, $maxtf$ 는 문서내에서 최대 색인어 빈도를 의미한다.

일반적으로 색인과정에서 색인어에 대한 가중치 계산을 수행한다. 하지만 이와 같이 검색시 가중치 계산을 수행하는 이유는 동적인 삽입/삭제를 수행하기 때문이다. 다시 말해서, 만약 색인시에 가중치를 계산한다면, 동적 문서 삽입/삭제를 수행할 때마다 모든 색인어의 가중치를 다시 계산해야 하는 오버헤드가 발생하기 때문이다.

질의 Q 와 문서집합 D 의 랭킹은 세가지 검색 모델 불리언 검색 모델, 확장 불리언 검색 모델, 벡터공간 모델을 변형하여 지원한다. 각각 변형된 모델은 다음과 같다.

불리언 검색 모델에서 문서의 랭킹은 다음과 같은 식에 의해 이루어진다. 문서집합의 총 개수인 n 차원의 벡터 W^B 는 다음과 같이 표현된다.

$$W^B = (w_j)_{j=1, n}$$

벡터의 원소인 w_j 는 문서 d_j 의 랭킹 값을 의미한다.

$$Q_{and} \text{ 일 경우, } w_j = \min(qw_1 dw_j, qw_2 dw_j)$$

$$Q_{or} \text{ 일 경우, } w_j = \max(qw_1 dw_j, qw_2 dw_j)$$

$$Q_{not} \text{ 일 경우, } w_j \text{는}$$

$$\text{if } (qw_i dw_j) > 0, \quad 0$$

$$\text{else, } \max_{i(qw_i = qw)}(qw_i dw_j, qw_i dw_j)$$

이다.

확장 불리언 검색 모델의 유사도 계산은 다음과 같은 식에 의해 이루어진다. 연산자의 강도를 나타

내는 계수 p 값은 가장 나은 성능을 보이는 값 2를 사용하였다. 문서집합의 총 개수인 n 차원의 벡터 W^E 는 다음과 같이 표현된다.

$$W^E = (w_j)_{j=1, n}$$

$$Q_{or} \text{ 일 경우, } w_j = \sqrt[p]{\frac{qw_1^p dw_j^p + qw_2^p dw_j^p}{qw_1^p + qw_2^p}}$$

Q_{and} 일 경우,

$$w_j = 1 - \sqrt[p]{\frac{qw_1^p(1 - dw_j^p) + qw_2^p(1 - dw_j^p)}{qw_1^p + qw_2^p}}$$

Q_{not} 일 경우, 는 $w_j = 1 - dw_j$

벡터 공간 모델에서의 문서의 랭킹은 다음 식에 의해 수행된다. 문서집합의 총 개수인 n 차원의 벡터 W^Y 는 다음과 같이 표현된다.

$$W^Y = (w_j)_{j=1, n}$$

$$w_j = qw_1 dw_j + qw_2 dw_j$$

4. 응용 분야

1) SGML 및 XML 문서 검색

SGML 문서와 XML 문서를 각각 파서를 이용하여 문서를 n 개의 논리적 부분(필드)로 나눈다면 교차 필드 검색을 수행할 수가 있다. 이러한 검색 엔진은 SGML과 XML 문서내의 모든 계층정보와 모든 엘리먼트 정보를 색인하는 구조 검색 엔진의 문제점인 색인공간이 상당히 소요되는 문제와 검색속도의 저하 문제를 해결할 수 있다[14].

2) HTML 문서 검색

웹 공간 및 유즈넷 공간의 검색이 인터넷 검색 엔진의 가장 주된 검색대상이라 할 수 있으므로 이 두 공간을 검색할 때 사용할 수 있는 필드들을 나열하면 다음과 같다.

- Web: title, link, head, first heading, url, host, anchor, text, site
- Usenet: author, subject, newsgroups, date, keyword, summary

3) 기타 분야

교차필드검색에 사용될 수 있는 기타 분야로는 다음을 예로 들 수 있다.

- 전화번호부: company name, phone number, address
- 기업일반현황정보: company name, sales, president, history, sic, address, phone number
- 뉴스정보: headline, by-line, body, date, source, section
- 특허정보: title, patent number, assignee name, application date, abstract
- 전시회정보: title, period, country, city, contact point, industry
- 단체정보: name, president, address, phone number, description, history, country, city

5. 사례연구: STEER-XDS

본 논문에서는 교차 필드 검색을 지원하는 검색 시스템으로 STEER-XDS(XML Document Searcher)의 구현 사례를 제시한다. 본 시스템은 한국전자통신연구원 컴퓨터.소프트웨어 기술 연구소 언어공학부에서 개발한 정보검색시스템으로 XML 문서 기반 정보 검색 시스템이다.

그림 5는 STEER-XDS의 검색 화면을 보여주고 있다. 사용자가 예측이라는 단어가 한글요약에 나오고 저자그룹에 유효열이 있는 문서를 찾고자 할 때, 질의입력 콤보박스에 “예측 in 한글요약 AND 유효열 in 저자그룹”을 입력한 예를 보여주고 있다.

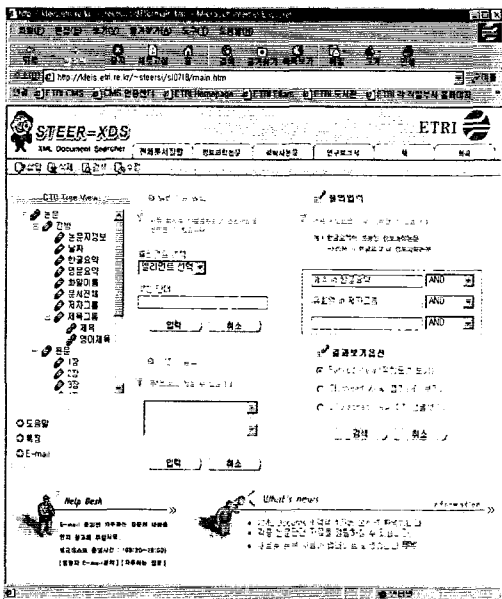


그림 5. STEER-XDS 검색화면

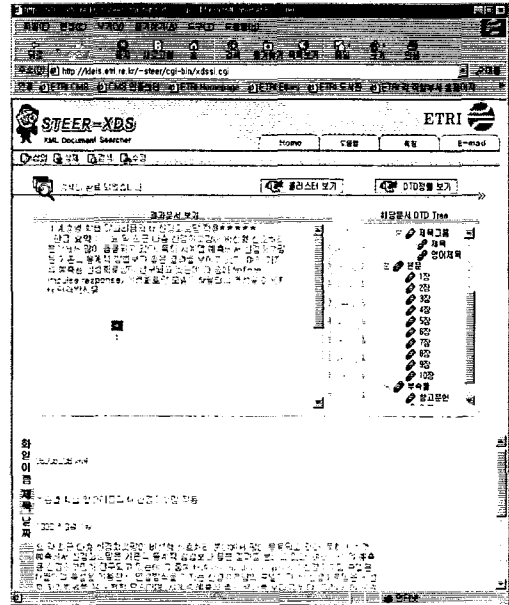


그림 6. STEER-XDS 결과화면

그림 6은 그림 5의 질의 예를 통해 검색한 결과 화면이다. 검색 결과인 문서의 부분과 전체를 사용자에게 동적으로 제시하는 화면이다.

STEER-XDS는 내용과 필드 구조를 이용하는 교차 필드 검색으로 다음과 같은 검색측면에서의 장점을 갖을 수 있다. 기존의 단순 색인어만으로 문서에 접근하는 기능 외에 문서의 필드 구조정보와 내용정보를 혼합하여, 다양한 각도로 문서를 검색할 수 있다. 예를 들어, 특정한 키워드가 제목에 포함되거나, 저자 소속, 요약 등의 문서의 특정 부분을 지칭하여 검색의 범위를 한정할 수 있는 검색이 가능하다.

아울러 문서의 일부만을 골라내어 자유롭게 조합하여 사용자에게 제시할 수 있다. 즉 문서전체를 보낼 필요없이 문서의 요약, 저자부분, 참고문헌만을 독립적으로 분리하여 볼 수 있다. 따라서 불필요한 네트워크의 과부하와 시스템의 과부하를 감소시킬 수 있다. 예를 들어, 1장만 분리하여 보여

출 수 있으며, 문서내에 포함된 그림만을 조합한 새로운 형태로 제시가 가능하여 제시된 문서의 재사용 또한 보장된다.

6. 결론

본 논문에서는 내용과 필드 구조를 통합하는 검색모델인 교차 필드 검색 모델을 제안하였다. 기존의 검색 시스템이 문서를 단지 단어의 연속이라는 제한적 관점에서만 바라보는 문제를 해결하였다. 아울러 기존의 필드 검색이 고정된 필드를 색인 및 검색대상으로 보거나, 단일 필드와 검색어를 통한 검색만을 지원하는 점을 보완하였다.

본 모델은 기존 정보검색 시스템의 기능을 기본으로 제공하고, 필드 단위 색인/검색 기능과 아울러 색인 구조는 동적인 문서 삽입/삭제 기능을 제공한다. 검색 측면에서는 질의를 파싱하고 난 후 문서 랭킹을 위해 세가지 검색 모델 불리언 검색 모델, 확장 불리언 검색 모델, 벡터 공간 모델을 선택할 수 있다.

또한 본 논문에서는 교차 필드 검색의 구현사례로 한국전자통신연구원 언어공학부에서 개발한 정보검색시스템인 STEER-XDS에 대해 알아보았다.

참고문헌

- [1] AltaVista <http://www.altavista.com>
- [2] Deja News <http://www.deja.com>
- [3] Digial LINNET <http://dengine.postech.ac.kr/DL>
- [4] Frakes, W. B., Baeza-Yates, R., Information Retrieval: Data Structures & Algorithm, Prentice-Hall, 1992.
- [5] Galaxy <http://www.galaxy.com>
- [6] HotBot <http://www.hotbot.com>
- [7] Infoseek <http://www.infoseek.com>
- [8] Lycos <http://www.lycos.com>
- [9] Open Text Power Search
<http://www.opentext.com>
- [10] Salton, G. Automatic Text Processing, Addison Wesley, 1989.
- [11] REHABDATA <http://welfare.or.kr/rehab/indx.htm>
- [12] Yahoo <http://www.yahoo.com>

- [13] 국회도서관 <http://www.nanet.go.kr>
- [14] 박영찬, 김문석, 김남일, "SGML/XML 정보검색 시스템의 구성과 구현 방법론 사례연구 : STEER-SGML/XML, 제 10 회 한글 및 한국어 정보처리 학술발표대회, pp. 105-110, 1998.
- [15] 박현주, 최재덕, 강상배, 박승, 박용욱, 권혁철, "인터넷 홈페이지 검색 시스템 구현과 검색효율 향상", 제 9 회 한글 및 한국어 정보처리 학술발표대회, pp. 227-232, 1997.
- [16] 송인석, 박혁로, "텍스트 이해 모델에 기반한 정보 검색 시스템", 제 9 회 한글 및 한국어 정보처리 학술발표대회, pp. 1-6, 1997.
- [17] 유럽사회과학정보협의회
<http://maeul.welfare.net/search/cessda.htm>
- [18] 윤보현, 김상범, 임해창, "한국어정보검색에서 구문적 용어불일치 완화방안", 제 10 회 한글 및 한국어 정보처리 학술발표대회, pp. 143-149, 1998.
- [19] 한글라이코스 <http://www.lycos.co.kr>
- [20] 한글알타비스타 <http://www.altavista.co.kr>
- [21] 한글야후 <http://www.yahoo.co.kr>