메타데이타와 텍스트 정보의 통합검색 모델
(A Hybrid Information Retrieval Model Using Metadata and Text)

유 정 목 † 맹 성 현 †† 김 성 수 ††† 이 만 호 ††††
(Jeong-Mok Yoo) (Sung-Hyon Myaeng) (Sung-Soo Kim) (Mann-Ho Lee)

요 약 메타데이타를 위한 검색모델은 질의에 사용자의 정보요구를 정확하게 반영하기 때문에 정확율(precision)은 높지만 질의 조건에 만족하지 않는 정보를 배제하므로 재현율(recall)은 낮다. 반면 전문(full-text) 텍스트 검색 모델은 사용자 질의에 대하여 모든 문서를 검색대상으로 하므로 정확율은 낮고 재현율은 높다. 메타데이타 검색모델의 높은 정확율은 사용자가 메타데이타의 구조적 특성에 맞게 질의를 구성할 경우 가능하지만 일반적으로 사용자가 메타데이타의 구조 정보를 반영한 사용자 질의를 구성할 수 있다고 기대하기는 어렵다. 또한 메타데이타에 포함된 정보의 양은 전문 텍스트가 가진 정보의 양보다 적기 때문에 텍스트를 검색한 결과보다 재현율이 떨어진다. 본 논문에서는 이러한 특성을 반영하여 메타데이타 검색 시, 사용자의 다양한 질의를 메타데이타의 특성에 맞게 재구성하고 메타데이타뿐 아니라 텍스트에 대해서도 검색을 수행하여 두 모델의 장점을 함께 고려한 통합 검색 모델을 제안한다.

키워드 : 메타데이터 검색 모델, 메타데이터 필드 확장, 구조 질의, 하이브리드 검색 모델

Abstract Metadata IR model has high precision and low recall because the query in Metadata IR model is strict that is, the query can express user information need exactly, while Full-text IR model has low precision and high recall because the query in Full-text IR model is a kind of simple keyword query which expresses user information need roughly. If user can translate one's information need into structured query well, the retrieval result will be improved. However, it is little possible to make relevant query without understanding characteristics of metadata. Unfortunately, most users do not interested in metadata, then they cannot construct well-made structured query. Amount of information contained in metadata is less than text information. In this paper, we suggest hybrid IR model using metadata and text which can provide users with lots of relevant documents by retrieving from metadata field and text field complementarily.

Key words : Requirements Change Management, Requirements Change Management Process, Software Product Lines

1. 서 론

메타데이타란 일반적으로 데이타에 관한 데이타로서 정보자원의 다양한 속성을 기술하는 부가적인 데이타를 의미한다. 즉 메타데이타란 물리적인 의미의 데이타(예를 들어, 비디오, 오디오, 텍스트 등)는 아니지만, 해당 데이타와 직간접적으로 관련된 정보를 제공하는 데이타를 의미한다. 이와 같은 메타데이타를 정보검색과 같은 응용 분야에서 사용하면 사용자가 원하는 정보를 좀 더 쉽고 빠르게 찾아낼 수 있다. 이러한 이유로 여러 응용 분야에서 다양한 메타데이타 표준들이 존재하며 현재 연구되고 있다. 그러나 응용 분야에서 메타데이타를 효율적으로 사용하기 위해서는 각 메타데이타의 구조적 특성을 반영한 사용자 질의를 구성하여야 한다. 즉, XML 형식의 메타데이타를 정의할 때 사용한 XML Schema 또는 XML DTD에 기술된 구조 정보를 반영한 사용자 질의 작성이 중요하다. 그러나, 일반적으로 사용자는 메타데이타의 복잡한 구조 정보를 이해하고, 그 특성을 반영한 사용자 질의를 구성하는 것에 익숙하지 않다. 따라서 사용자가 원하는 정보 요구를 적절히

† 정 회 원 : 한국전자통신연구원 디지털홈연구단 인터넷서버그룹 연구원
jeongmok@gmail.com
†† 종신회원 : 한국정보통신대학교 공학부 교수
myaeng@icu.ac.kr
††† 정 회 원 : 한국통신 비지니스부문 프로젝트 관리부 사원
kss@icu.ac.kr
†††† 종신회원 : 충남대학교 전기정보통신공학부 교수
mhlee@cnu.ac.kr
(Corresponding author임)
논문접수 : 2006년 2월 24일
심사완료 : 2007년 3월 20일

파악하여 메타데이타의 구조적 특성에 맞게 사용자 질의를 재구성하는 작업이 필요하다.

앞에서 언급한 바와 같이 현재 다양한 형식의 데이타를 위한 메타데이타 표준들이 존재한다. 본 논문에서는 그 중에서 텍스트 데이타를 위한 메타데이타를 대상으로 정보검색 응용 분야에서 사용자에게 좀 더 효율적인 검색 결과를 제공해 줄 수 있는 방안에 대해 기술한다. 텍스트를 위한 메타데이타는 텍스트 자원의 다양한 부가 속성을 기술하였기 때문에 자원 자체, 즉 텍스트보다는 소량의 정보를 가지고 있다. 이러한 한계점으로 인해, 메타데이타 내 기술되지 않은 정보를 찾는 사용자 질의를 이용하여 검색 서비스를 요청한 경우 적합한 정보를 찾을 수 없다. 반면, 메타데이타를 이용하여 부가 정보를 기술한 자원인 텍스트 데이타는 메타데이타보다 양적으로 더 많은 정보를 가지고 있다. 사용자가 검색 대상을 메타데이타뿐만 아니라 텍스트 데이타로 확장하여 검색할 경우 사용자가 원하는 정보를 찾을 확률이 높아진다.

그림 1은 특정 질의에 대한 텍스트 문서 적합성 분포 다이어그램이다. 사용자가 텍스트 데이타와 텍스트 데이타에 대한 메타데이타를 대상으로 검색 서비스를 수행할 경우, 실제 사용자 질의에 적합한 문서는 메타데이타 내에 존재할 수도 있으며, 텍스트 데이타 내에 존재할 수도 있다. 또한, 사용자가 메타데이타에 대한 질의를 작성할 때 지정한 메타데이타 필드 외에 다른 메타데이타 필드에 존재할 가능성도 있음을 보여준다.

그림 1의 다이어그램을 참고해 볼 때 세 가지 경우를 고려해 볼 수 있다. 첫째, 정보검색 시스템에서 메타데이타 특정 필드만을 대상으로 검색하였을 경우 사용자가 지정한 필드만을 이용함으로써 사용자에게 제한된 검색 결과만을 제공할 수가 있다. 둘째, 텍스트만을 대상으로 시스템이 검색할 경우 메타데이타에 기술된 부가 정보들을 제외한 텍스트 부분의 정보만을 제공할 수 있다. 마지막으로, 메타데이타 특정 필드와 메타데이타

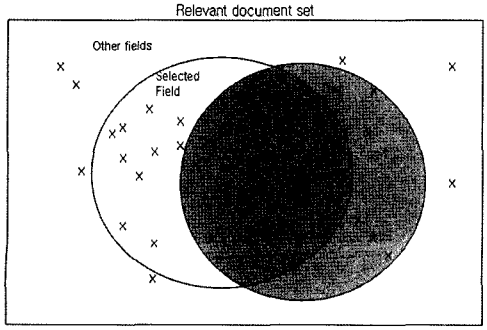


그림 1 적합문서 분포 다이어그램

가 설명하는 텍스트를 함께 검색하는 경우, 앞에서 언급한 두 가지 경우보다 더 많은 검색 결과들을 사용자에게 제공할 수 있다. 이를 위해서는 사용자는 메타데이타 필드들 중에서 사용자 질의에 가장 적합한 필드를 선택하여 질의에 이용하고, 또한 메타데이타 검색 결과를 보완해주기 위해 텍스트 데이타를 검색에 직접 이용하는 방법이 필요하다.

그러나 다양한 응용 분야에서 복잡한 구조 정보를 가진 메타데이타 표준이 연구되고 제안되는 현 시점에서 개별적인 메타데이타의 구조적 특성을 일일이 사용자가 파악하여 사용자 질의를 작성하는 것은 매우 어려운 일이다. 따라서 본 논문에서는 사용자의 키워드 기반 질의어들을 메타데이타 검색에 적합한 사용자 질의로 자동적으로 변환하여 메타데이타를 대상으로 검색할 수 있는 방법을 제시한다. 사용자가 작성한 키워드 집합과 메타데이타 필드 집합 사이의 유사도 값을 이용하여 사용자가 이용한 검색 키워드들과 메타데이타 필드들 사이의 상호 연관성을 검색에 반영하여 일반적으로 사용하는 키워드 기반 질의어들을 이용하여 메타데이타 구조 정보를 반영한 사용자 질의를 자동 생성한다.

또한 텍스트 부분을 메타데이타를 구성하는 메타데이타 필드들과 함께 상호 보완적으로 검색하여 메타데이타만을 대상으로 한 검색 결과를 보완하여 좀 더 향상된 검색 결과들을 사용자에게 제공하는 모델을 제안한다.

2. 관련연구

키워드 중심의 질의를 메타데이타의 특성에 맞게 변환하는 방안은 이전에 연구되었다. [5]에서는 단순히 키워드 중심으로 구성된 질의를 비구조형 질의(Unstructured query), 메타데이타 필드와 키워드로 구성된 질의를 구조형 질의(Structured query)라고 정의하였고 구조형 질의를 이용한 검색이 비구조형질의를 이용한 검색보다 우수한 결과를 보인다는 것을 증명하였다.

Goncalves et al[8]에서는 [5]에서 언급한 내용들을 기반으로 하여 Bayesian network 모델 기법을 응용하여 비구조형 질의를 구조형 질의로 자동적으로 변환하는 방안을 제시하였다. [8]에서는 사용자의 질의와 문서의 모든 메타데이타 필드를 조합함으로써 가능한 모든 구조형 질의 후보군을 생성한다. 각 후보질의들이 생성될 수 있는 확률을 Bayesian network 모델 기법을 이용하여 계산하고 확률이 높은 상위 5개의 구조형 질의만을 사용자의 정보요구에 부합하는 질의로 가정한다. 이 상위 5개의 구조형 질의를 이용하여 메타데이타 검색을 수행하고 추출된 결과를 통합한다. [8]에 관한 주요 알고리즘은 유사연구비교(5장)에서 자세히 설명한다.

[8]에서 사용한 Bayesian network 모델은 Turtle &

Croft[14]에 의해 처음으로 정보검색 모델에 응용되었으며 후에 Ribeiro-Neto & Muntz[13]에 의해 발전되었다. Bayesian network은 문서간의 순위를 결정하는 기법뿐 아니라 relevance feedback[9], query expansion [4], information filtering[2], classification[3,16], SGML 구조문서 검색[10] 등 다양한 정보검색 분야에 응용되고 있다.

구조 문서에 대한 검색과 질의를 위해 다양한 검색 모델들과 질의 언어들에 대한 연구가 지속적으로 연구되고 있다[17-20]. 그러나, 본 논문에서는 구조 문서가 아닌 메타데이타를 그 대상으로 하고 있으며, Bayesian network 모델을 이용하여 메타데이타 검색에 제일 적합한 사용자 질의를 작성하여 메타데이타에 대한 검색 효율을 증진하는 접근 방법에 대해 가장 참고할 만한 관련 연구 내용은 [5,8]이다.

3. 정 의

본 논문에서의 검색대상은 메타데이타(M_i)와 텍스트(T_i)로 구성되거나 혹은 메타데이타 만으로 구성된 문서(d_i) 들이며 각 문서들은 문서집합(D)를 구성한다.

$$D=\{d_1,d_2,...,d_i,...,d_n\} \quad n\geq 1$$

$$d_i=<M_i,T_i> \quad T_i=\varnothing \text{ or } T_i=<t_1,...,t_j>$$

d_n은 컬렉션 D를 구성하는 n번째 문서를 의미한다. 문서를 구성하는 메타데이타(M_i)는 한 개 이상의 메타데이타 필드(F_k)와 필드값(v_{ki})을 갖는다. 또한 문서가 텍스트 필드를 갖는다면 텍스트는 한 개 이상의 용어(t_i)로 구성된다.

$$M_i=<(F_1,v_{1i}),(F_2,v_{2i}),....,(F_k,v_{ki})> \quad k\geq 1,$$

$$\text{IF } T_i\neq\varnothing,\ T_i=<t_1,...,t_i,..,t_l> \quad l>1$$

예를 들어 특정 문서 d_i이 title, author, publisher의 메타데이타 필드로 구성되며 각각 필드값으로서 "난중일기", "이순신", "민음사"를 갖는다면, d_i=<(title, "난중일기"), (author, "이순신"), (publisher, "민음사")>로 나타낼 수 있다.

질의는 비구조형 질의(UQ, Unstructured Query)와 구조형 질의(SQ, Structured Query)로 분류한다. 비구조형 질의는 키워드로만 구성된 질의이며 구조형 질의는 메타데이타에 대한 질의(Q_M)와 텍스트에 대한 질의(Q_T)로 이루어진다.

$$UQ=<t_1,t_2,..,t_i,..,t_l> \quad l\geq 1$$

$$SQ=<Q_M,Q_T>$$

메타데이타에 대한 질의(Q_M)는 메타데이타 필드(F_i)와 필드값(v_{iq})의 쌍으로 구성된 질의이고 텍스트에 대한 질의(Q_T)는 비구조형 질의(UQ)와 같다. 또한 각 질의 값(v_{iq})은 하나 이상의 용어(t_i)로 구성된다. 이들은 아래와 같이 표현한다.

$$Q_M=<(F_1,v_{1q}),(F_2,v_{2q}),..,(F_i,v_{iq}),..,(F_m,v_{mq})> \quad m\geq 1$$

$$Q_T=UQ$$

$$v_{iq}=<t_1,t_2,..,t_i,..,t_l> \quad l\geq 1$$

예를 들어 질의가 단순히 키워드들로 구성되어 "이순신, 한산도 대첩, 임진왜란"이라면 비구조형 질의는 UQ =<"이순신, 한산도 대첩, 임진왜란">으로 표현할 수 있다. 질의가 메타데이타 필드와 짝을 이루어 제목은 "난중일기", 저자는 "이순신"이라고 하면, 구조형 질의로써 SQ=<(title, "난중일기"), (author, "이순신")>로 표현할 수 있다.

본 논문에서는 구조형 질의를 구성하는 필드를 크게 두 가지로 나눈다. 하나는 사용자가 지정한 필드(F_{user_select})이며 또 다른 하나는 시스템이 지정한 필드(F_{system_select})이다. 만일 사용자가 구조형 질의를 SQ=<(title, "난중일기"), (author, "이순신")>와 같이 작성하였다면 F_{user_select}는 title과 author필드이다. 또한 사용자가 비구조형 질의를 한 경우 자동적으로 구조형 질의로 변환이 되는데 이때 변환된 필드는 F_{system_select}에 해당된다.

4. 검색환경

본 논문에서 다음과 같이 가정한다.

첫째, 사용자는 비구조형 질의로도 질의할 수 있으며 구조형 질의로도 질의할 수 있다.

둘째, 사용자는 구조형 질의를 효율적으로 구성하지 못할 수 있다. 따라서 사용자가 선정한 메타데이타 필드보다 질의에 더 적합한 메타데이타 필드가 존재할 가능성이 있다.

셋째, 메타데이타 필드 검색만으로는 충분한 정보를 사용자에게 제공할 수 없으며 따라서 텍스트를 보완적으로 검색한다.

본 논문에서 제안하는 모델은 아래와 같은 절차로 진행된다.

1. 가장 질의에 적합한 메타데이타 필드선정: 사용자 질의와 메타데이타 필드간의 적합성을 측정한다.

측정한 적합성을 바탕으로 가장 적합성이 높은 메타데이타 필드를 선정하여 구조형 질의로 변환한다. 예를 들어 UQ=<"이순신">이며 "이순신"이라는 질의가 author필드와 가장 적합하다면 SQ=<(author, "이순신")>으로 전환된다.

2. 선정된 필드에 대하여 메타데이타 검색 수행: SQ=<(author, "이순신")>인 경우, author필드 내에 "이순신"을 갖는 문서와 유사도를 측정한다.
3. 텍스트에 대하여 병행 검색 수행: SQ = <(text, "이순신")>의 질의를 구성하여 검색한다.
4. 메타데이타 필드 검색과 텍스트 검색을 수행한 결과에 대하여 하나의 유사도로 통합 후 순위 결정: SQ=<(author, "이순신")>로부터 얻은 결과와 SQ = <(text, "이순신")>로부터 얻은 결과를 하나의 문서 유사도로 통합하여 순위를 결정한다.

5. 유사 연구 비교

본 연구의 목표는 사용자가 메타데이타의 특성을 이해하지 못하는 경우에도 가장 적합한 질의를 자동 생성하여 검색하며 텍스트 필드를 상호 보완적으로 검색하여 질의와 가장 적합한 문서를 찾아내는 것이다. 관련연구에서 이미 언급하였듯이 사용자의 다양한 질의를 자동적으로 구조형 질의로 전환해주는 연구는 Goncalves et al[8]에서 먼저 수행되었다. 따라서 본 논문에서는 [8]에서 제안된 방안을 실험을 통해 본 연구에서 제안하는 방안과 비교하였으며 그들의 제안방안을 간단히 VT(Virginia Tech)로 명기하였다.

5.1 질의 처리

VT에서 질의 처리는 다음과 같은 세 단계로 나뉜다.

1) 사용자의 비구조형 질의를 입력 받는다.
2) 모든 조합 가능한 구조형 질의를 구성한다.
3) 후보 구조형 질의들의 순위를 결정한다.

단계별 처리과정을 자세히 설명하기 위하여 먼저 컬렉션을 구성하는 문서는 title과 author로 구성되며 초기 비구조형 질의가 $UQ=<t_1,t_2,t_3>$라 가정한다. 후보 질의들을 구성하기 위해 문서를 구성하는 메타데이타 필드들과 용어들의 쌍의 형태로 조합한다. 만일 t_1이 tile과 author에 나타나며 t_2,t_3는 title에만 나타난다면 메타데이타 필드와 용어로 이루어진 쌍의 구성은 (author, t_1), (title, t_1)과 (title, t_2), (title, t_3)가 된다. 이를 바탕으로 후보질의는 다음과 같이 구성된다.

$Q_1 =< (author,t_1),(title,t_2),(title,t_3) >,$

$Q_2 =< (title,t_1),(title,t_2),(title,t_3) >$

후보 질의들은 문서가 갖는 메타데이타 필드 수와 질의의 양에 따라서 매우 다양하게 나타날 수 있다. VT에서는 이러한 후보질의들에 순위를 할당하여 상위 5개의 질의만을 검색에 적합한 구조형 질의로 가정한다.

5.2 후보질의 간의 순위 결정

VT에서는 후보질의 간의 순위를 결정하기 위하여 Bayesian network 모델을 이용한다. 설명을 용이하게 하기 위하여 문서는 두 개의 필드로 구성되어 있다고 가정하며 VT 모델은 그림 2처럼 표현할 수 있다.

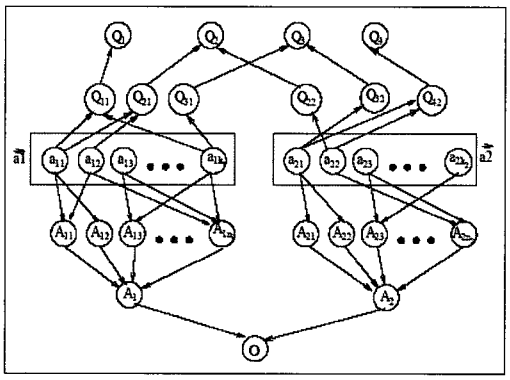


그림 2 후보질의 순위 결정을 위한 Bayesian network 모델

그림 2는 문서 집합을 구성하는 특정 문서(O)가 두 개의 메타데이타 필드(A_1, A_2)로 구성되며 각기 여러 개의 필드값($A_{11} \sim A_{1n}, A_{21} \sim A_{2n}$)들로 구성됨을 보여준다. 필드값들은 다양한 용어들($a_{11} \sim a_{1k_1}, a_{21} \sim a_{2k_2}$)로 구성되며 이러한 메타데이타 필드에 나타날 수 있는 모든 용어들은 하나의 벡터($\overrightarrow{a1}, \overrightarrow{a2}$)를 이룬다.

Q_i는 후보 구조형 질의들을 의미하며 Q_{ij}는 메타데이타 필드 j에 관한 질의이다. 그림 2에서 Q_1은 하나의 메타데이타 필드에 대해서 질의를 하며 Q_2는 두 개의 메타데이타 필드에 대하여 질의를 한다. 메타데이타에 대한 질의 Q_{ij}는 다양한 용어($a_{11} \sim a_{1k_1}, a_{21} \sim a_{2k_2}$)들을 질의에 이용한다.

후보질의들의 우선순위는 문서에 대한 컬렉션 O가 주어졌을 때 Q_i가 구성될 수 있는 확률($P(Q_i \mid O)$)로 결정된다. $P(Q_i \mid O)$는 구조질의를 구성하는 메타데이타 필드의 필드값들과 Q_{ij}에 나타나는 질의값들, 즉 활성화된 질의 벡터내의 질의값들과 코사인 유사도를 구한 후 일반화하여 메타데이타 필드와 질의와의 적합성을 계산

한다. $P(Q_i \mid O)$ 의 공식은 아래와 같다.

$$P(Q_i \mid O) = \eta * \frac{1}{2}[1 - \prod_{j=1}^{n_1}(1-\cos(A_{1j}, \vec{a_1})) + 1 - \prod_{j=1}^{n_2}(1-\cos(A_{2j}, \vec{a_2}))]$$

n_1과 n_2는 필드를 구성하는 용어의 개수를 나타내며 VT에서 지정한 η는 상수이다.

VT에서는 결정된 후보질의들 중 가장 우수한 질의를 검색에 이용하는 방안(1순위의 질의를 검색에 활용한다는 의미에서 VT1으로 명기한다.)과 상위 5개의 질의를 이용하여 결과를 통합하는 방안(VT1~VT5)을 연구하였다

본 논문에서는 VT1과 VT1~VT5의 결과와 본 논문에서 제안하고자 하는 방안을 실험을 통하여 비교한다.

6. 통합 검색 모델

6.1 구조형 질의 처리

본 논문에서는 사용자가 구조형 질의를 효율적으로 작성하기가 어렵다고 가정하였고 질의에 더 적합한 메타데이타 필드가 존재할 수 있다고 가정하였다. 따라서 사용자가 선정한 메타데이타 필드보다 더 적합한 필드가 있는지 확인하여야 한다.

먼저 사용자가 지정한 필드와 질의간의 적합성을 측정한다. 즉, SQ=<(title, "information retrieval"), (author, "김")>인 경우에 사용자가 지정한 필드 F_{title}과 "information retrieval"의 조합이 적절한지, F_{author}와 "김"의 조합이 적절한지를 판단하여야 한다.

VT에서는 Bayesian network 모델을 활용하였으나 본 논문에서는 이 적합성의 판단 기준으로 벡터 모델의 유사도[1] 개념을 채택하였다. 두 벡터간 유사도를 측정하기 위해서는 문서 벡터와 질의 벡터, 두 벡터가 필요하다. 이를 위하여 컬렉션 내의 모든 문서들의 용어들을 필드를 기준으로 재구성하여 가상의 문서(C_{field})를 생성하였다. 즉 C_{title}은 컬렉션내의 title필드에 나타나는 모든 용어들로 구성된 가상의 문서이다. 가상의 문서 C_{field}와 질의간의 유사도는 아래와 같이 표현한다.

$$Sim(\vec{C}_{field}, \vec{q})$$

사용자가 지정한 필드(F_{user_select})에 나타나는 용어로 구성된 가상의 문서는 C_{user_select}라고 정의하며 시스템이 지정한 필드(F_{system_select})에 나타나는 용어로 구성된 가상의 문서는 C_{system_select}라고 정의한다.

$\vec{C}_{user_select}$는 사용자가 지정한 필드에 나타나는 모든 용어들로 구성된 가상 문서의 벡터를 의미하며, $\vec{q}$란 구조형 질의를 이루는 필드값 벡터를 의미한다. 예를 들어 SQ=<(title, "information retrieval")>인 경우 title과 "information retrieval"과의 적합성은 가상의 문서 C_{title}와 질의벡터 "information retrieval"과의 유사도로서 측정된다.

사용자는 잘못된 구조형 질의를 구성할 수 있다고 가정하였으므로, 사용자가 지정하지 않은 필드들과 질의간의 적합성을 측정한다. 사용자가 지정한 메타데이타 필드를 제외한 나머지 메타데이타 필드들을 대상으로 질의와의 적합성을 측정한 후, 이들 중 가장 유사도가 높은 메타데이타 필드를 적합한 필드로 가정하며 가장 적합성이 높은 필드가 시스템이 지정한 필드이다. (F_{system_select})

$$Sim(\vec{C}_{system\_select}, \vec{q}) = \max[Sim(\vec{C_1}, \vec{q}), ..., Sim(\vec{C_i}, \vec{q})]$$
$$i \neq user\_select$$

예를 들어 SQ=<(title, "information retrieval")>인 경우 title을 제외한 나머지 필드들과 "information retrieval"과의 적합성을 측정한다. 나머지 필드들과 적합성을 측정하여 가장 높은 적합성을 보인 필드가 F_{system_select}가 되며 가장 높은 적합성을 시스템이 지정한 필드와 질의간의 적합성으로 간주한다.

이러한 경우 다음과 같이 두 가지 경우가 발생한다.

1) $Sim(\vec{C}_{system\_select}, \vec{q}) \leq Sim(\vec{C}_{user\_select}, \vec{q})$

2) $Sim(\vec{C}_{system\_select}, \vec{q}) > Sim(\vec{C}_{user\_select}, \vec{q})$

첫 번째, 사용자가 지정한 메타데이타 필드와 질의간의 적합성이 다른 메타데이타 필드와 질의간의 적합성에 비교해 볼 때 더 높은 적합성을 가진다고 생각할 수 있다. 따라서 사용자는 적합한 구조형 질의를 생성하였다고 볼 수 있다. 그러나 두 번째 경우와 같이 사용자가 지정한 메타데이타 필드와 질의간의 적합성보다 그 외 메타데이타 필드와 질의간의 적합성이 더 높으므로, 사용자가 올바르게 구조형 질의를 구성하였다고 볼 수 없다. 그러므로, 사용자가 작성한 구조형 질의보다 시스템에서 추천한 구조형 질의를 이용하여 검색 서비스를 제공하는 것이 더 적합하다는 것을 알 수 있다.

메타데이타를 구성하는 메타데이타 필드와 질의 사이의 적합성을 판별하여 어떤 메타데이타 필드를 구조형 질의에 사용할 것인지 결정하였다. 앞에서 언급한 바와 같이, 본 논문에서는 메타데이타 필드뿐만 아니라 메타데이타가 부연 설명하는 텍스트 데이타를 검색 대상에 포함하여 더 넓은 검색 서비스 범위를 제공한다. 이를 위해, 텍스트 데이타를 다른 메타데이타와 같은 하나의

메타데이타 필드로 정의하고 사용자 질의의 대상으로 간주한다.

텍스트 데이타에 대한 검색을 메타데이타에 대한 검색과 함께 수행하는 경우, 앞에서 선택한 메타데이타 필드 외에 추가 검색 대상 필드는 1)의 경우 사용자가 지정한 필드와 텍스트 필드이며, 2)의 경우는 사용자가 지정한 필드와 시스템이 추천한 필드, 그리고 텍스트 필드이다. 앞에서 언급한 바와 같이 시스템이 지정한 필드의 적합성이 사용자의 필드보다 높음에도 불구하고 사용자가 지정한 필드를 함께 검색하는 이유는 사용자의 정보요구를 반영하기 위해서 이다.

구조형 질의가SQ= <(title, "information retrieval"), (author, "김")>일 경우를 가정해보자. 먼저 가상의 문서 C_{title}와 질의 벡터 "information retrieval"과의 유사도를 측정하여 적합성을 판단하고 C_{author}와 질의 벡터 "김"과의 유사도를 측정하여 적합성을 판단한다. 만일 "김"이 "김의 전쟁"과 같이 author 필드가 아닌 title 필드에 더 적합한 경우가 발생한다면 사용자가 지정한 필드, 시스템이 지정한 필드, 그리고 텍스트 필드가 검색대상으로 선정되어 최종적으로 SQ= <(title, "information retrieval"), (text, "information retrieval"), (author, "김"), (title, "김"), (text, "김")>으로 작성된다.

6.2 비구조형 질의 처리

비구조형 질의일 경우 시스템이 구조형 질의를 생성해주어야 한다. 비구조형 질의인 경우, 사용자가 메타데이타 필드를 지정하지 않았으므로 모든 메타데이타 필드를 검색대상으로 하며 이들 중 가장 질의와 적합성이 높은 필드를 선정하여 구조형 질의를 생성한다. 따라서 검색 대상은 시스템이 추천한 메타데이타 필드(F_{system_select})와 텍스트 필드이다.

$$Sim(\overline{C}_{system\_select}, \overline{q}) = \max[Sim(\overline{C_1}, \overline{q}), ..., Sim(\overline{C_i}, \overline{q})]$$

만일 UQ = <information, retrieval, 김>이라면 UQ를 구성하는 질의값 3개에 대하여 각각 필드간의 적합성을 판단한다. "information"과 "retrieval" 둘 다 title 필드에 가장 적합하다면 (title, "information retrieval")로 구성되며 "김"이 author에 적합하다면 (author, "김")으로 전환되어 시스템이 지정한 필드와 텍스트 필드가 검색대상으로 선정되어 최종적으로는 SQ = <(title, "information retrieval"), (text, "information retrieval"), (author, "김"), (text, "김")>으로 구성된다.

6.3 질의와 필드간의 적합성 측정

질의와 필드간의 적합성 측정은 벡터 모델[1]을 기반으로 하였다. 벡터 모델은 문서 또는 질의에서의 중요도에 따라 추출된 용어들에 가중치를 부여함으로써, 문서와 질의를 가중치가 부여된 용어들의 벡터로 표현한다. 가상의 문서 C_{field}와 질의와의 유사도는 C_{field}에 출현하는 가중치가 부여된(w_{ik}) 용어의 벡터와 질의에 출현하는 가중치가 부여된 질의 벡터($g_k(\overrightarrow{q})$)의 내적으로 표현된다. 각 용어에 대한 가중치는 C_{field}에 대한 중요도를 반영하며 특정 필드 내에 자주 나타나고 전체 필드 중에서 적은 수의 필드에 출현하는 용어에 보다 높은 가중치를 부여한다. 또한 가상의 문서 C_{field}는 각각 벡터의 길이가 다르므로 정규화 요소($\sqrt{\sum_{t_k \in T_i} w^2_{ik}}$, $\sqrt{\sum_{t_k \in T_i} g_k(\overrightarrow{q})^2}$)로서 벡터의 길이를 일치시켜 공정한 유사도를 측정할 수 있다.

질의와 필드간의 적합성을 측정하는 방식은 아래와 같다.

$$Sim(\overrightarrow{C_i}, \overrightarrow{q}) = \cos(\overrightarrow{C_i}, \overrightarrow{q}) \cong \frac{\sum_{t_k \in T_i} w_{ik} g_k(\overrightarrow{q})}{\sqrt{\sum_{t_k \in T_i} w^2_{ik}} * \sqrt{\sum_{t_k \in T_i} g_k(\overrightarrow{q})^2}},$$

$$w_{ik} = ftf_i(k) * fidf(k)$$

여기서 $ftf_i(k)$는 메타데이타 필드 i내에서 나타나는 용어 k의 빈도수를 의미하며 $fidf(k)$는 용어 k가 나타나는 필드의 역수를 말한다. 메타데이타 필드 i는 컬렉션을 구성하는 문서들의 i번째 메타데이타 필드에 나타나는 용어들로 구성된다. $g_k(\overrightarrow{q})$는 질의 벡터내의 용어 k의 가중치를 의미한다. T_i는 i번째 필드F_i내에 존재하는 모든 용어의 집합이다.

6.4 질의와 선정된 필드를 갖는 문서들과의 유사도 측정

6.3에서 설명한 바와 같이, 벡터 모델[1]에서 제시한 유사도 공식을 이용하여 질의와 필드간의 적합성을 측정한 후 선정된 필드에 대하여 선정된 필드를 갖는 문서들을 검색한다. 예를 들어 SQ=<(title, "이순신")>이라면 title필드를 갖는 모든 문서가 검색대상이며 title필드 내에 질의어 "이순신"을 갖는 문서들을 검색한다. 선정된 필드에서 검색된 문서들과 질의어 사이의 유사도를 측정하기 위해서는 선정된 필드(F_i)에 질의어가 출현하는 문서(d_{ij})의 벡터와 질의벡터($\overrightarrow{q}$)가 필요하며 두 벡터간의 유사도를 측정한다. 6.3에서 언급한 바와 같이 특정 문서 내 특정 필드에 자주 나타나는 용어에 대하여 보다 높은 가중치를 부여하고 각 문서내의 필드의 벡터 길이를 정규화하여 공정한 유사도를 측정한다.

선정된 필드를 갖는 문서들과 질의간의 유사도 측정

공식은 아래와 같다.

$$Sim(\vec{d}_{ij}, \overrightarrow{q}) = \cos(\vec{d}_{ij}, \overrightarrow{q}) \cong \frac{\sum_{t_k \in T_i} w_{ik} g_k(\overrightarrow{q})}{\sqrt{\sum_{t_k \in T_i} w^2_{ik}} * \sqrt{\sum_{t_k \in T_i} g_k(\overrightarrow{q})^2}},$$

$$w_{ik} = ftf_{ij}(k) * idf_i(k)$$

$ftf_{ij}(k)$는 j번째 문서의 i번째 필드내의 용어 k의 빈도수이다. $idf_i(k)$는 i번째 필드에서 용어 k가 나타나는 문서들의 역수를 의미하며 두 변수를 곱하여 가중치를 계산한다.

6.5 문서 우선순위 결정

6.1과 6.2에서 기술하였듯이 본 논문에서는 하나의 질의에 대하여 비구조형 질의의 경우에는 2개의 필드(시스템이 지정한 필드와 텍스트 필드), 구조형 질의의 경우에는 2개 혹은 3개의 필드(사용자가 지정한 필드, 시스템이 지정한 필드, 그리고 텍스트 필드)를 검색 대상으로 한다. 따라서 각 필드를 검색한 결과를 통합하여 유사도를 기준으로 검색 순위를 결정할 수가 있다.

그림 3은 질의에 대하여 적합성을 판단하여(적합성 판단은 6.3 참조.) 적합하다고 판단된 필드 F_1과 필드 F_2에 대하여 검색을 수행하고 F_1에 질의가 출현하는 문서 d_{11}, d_{12}, d_{13}가 검색(필드내 문서와 질의간의 유사도 측정은 6.4 참조.) 되며 F_2에 질의가 나타나는 문서 d_{21}, d_{23}, d_{24}가 검색되어 이러한 결과 순위들을 통합한 하나의 문서 유사도를 생성하는 과정을 보여준다.

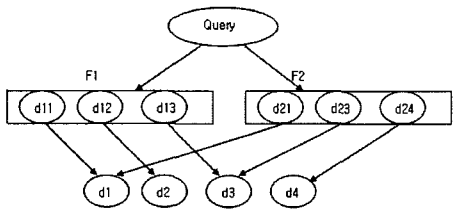


그림 3 필드검색 결과들의 통합

예를 들어 d_{11}의 유사도가 0.2이고 d_{21}의 유사도 또한 0.2라면 평균 처리할 경우 d_1의 유사도는 (0.2+0.2)/2이다. 그러나 첫 번째 필드에 나타나는 용어로 구성된 가상문서 C_1과 질의간의 유사도는 $Sim(\vec{C}_1, \overrightarrow{q})$=0.5이며 두 번째 필드에 나타나는 용어로 구성된 가상문서 C_2와 질의간의 유사도는 $Sim(\vec{C}_2, \overrightarrow{q})$=0.1 이라면 d_{11}과 d_{21}의 유사도가 같더라도 d_{11}의 유사도에 가중치를 부여하여 통합하는 것이 타당할 것이다. 즉, 메타데이타 필드와 질의간의 적합성이 높으면 높을수록 보다 많은 가중치를 부여하여 통합 시 이러한 특성을 반영해야 한다. 따라서 본 논문에서는 통합 시에 Bayesian network의 link matrix[13] 개념을 응용하여 통합한다. Bayesian network란 변수를 표현하는 노드와 변수들간의 의존관계를 표현하는 호(arc)의 방향성 비순환 그래프이다. 노드 P에서 노드 Q까지 호가 있다면 P는 Q의 부모노드라 부른다. 부모노드가 자식노드에 미치는 영향은 조건부 확률로서 표현하고 이러한 부모 노드들이 자식노드에 영향을 미치는 확률을 행렬로서 표현한 것이 link matrix이다.

자식노드 C에 부모노드 S_1, S_2로 연결되어 있다면 아래와 같은link matrix를 가지며 이에 대한 통합값을 구할 수 있다.

$$L_C = \begin{bmatrix} P(\neg C|\neg S_1,\neg S_2) & P(\neg C|S_1,\neg S_2) & P(\neg C|S_1,\neg S_2) & P(\neg C|S_1,S_2) \\ P(C|\neg S_1,\neg S_2) & P(C|S_1,\neg S_2) & P(C|S_1,\neg S_2) & P(C|S_1,S_2) \end{bmatrix}$$

$$\begin{aligned} B(C) = {} & P(C|\neg S_1,\neg S_2) * B(\neg S_1) * B(\neg S_2) \\ & + P(C|\neg S_1,S_2) * B(\neg S_1) * B(S_2) \\ & + P(C|S_1,\neg S_2) * B(S_1) * B(\neg S_2) \\ & + P(C|S_1,S_2) * B(S_1) * B(S_2) \end{aligned}$$

구체적인 예로 노드 Q가 A, B, C라는 부모노드를 가지며 부모노드들은 각기 아래와 같은 값을 갖는다고 가정하자.

$$P(A = true) = a, \quad P(B = true) = b, \quad P(C = true) = c$$

OR 결합의 경우, A, B, C 중 하나라도 true라면 Q는 true이며 A, B, C 모두 false일 경우만이 Q는 false가 된다. 따라서 이것은 아래와 같은 link matrix를 갖는다.

$$L_{OR} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

위 link matrix의 상단은 Q가 false일 때 이며 하단은 true일 경우를 의미한다. 각각의 컬럼에 해당하는 값들은 부모노드들의 값의 조합에 해당한다. 따라서 첫번째 컬럼 0 (000_2)은 A, B, C모드 false일 때이며, 두번째 컬럼 1(001_2)은 A와 B는 false이고 C만 true일 경우이다. Q가 true일 때의 값은 부모노드의 확률들과 하단의 열들의 값들과 곱함으로써 얻을 수 있다.

Q가 true일 경우를 계산하면 아래와 같다.

$$\begin{aligned} & P(Q = true) = (1-a)(1-b)c + (1-a)b(1-c) + (1-a)bc \\ & + a(1-b)(1-c) + a(1-b)c + ab(1-c) + abc \end{aligned}$$

이러한 link matrix를 이용하기 위하여 메타데이타 필드와 질의간의 적합성들을 link matrix로 간주한다. 즉, $P(F_i|Q) = Sim(\vec{C}_1, \vec{q})$와 $P(F_2|Q) = Sim(\vec{C}_2, \vec{q})$로 가정한다. 따라서 그림 [3]의 d_{11}과 d_{21}을 하나의 d_1으로

통합하기 위해서 아래와 같은 link matrix를 만들 수 있으며 이를 이용하여 통합 유사도를 구할 수 있다.

$$L_D = \begin{bmatrix} 1 & 1-Sim(\vec{C}_2,\vec{q}) & 1-Sim(\vec{C}_1,\vec{q}) & 0 \\ 0 & Sim(\vec{C}_2,\vec{q}) & Sim(\vec{C}_1,\vec{q}) & 1 \end{bmatrix}$$

하단의 첫째 컬럼 0은 두 개의 필드가 모두 존재하지 않을 때이며 두번째 값은 F_2만이 존재할 때, 네번째 컬럼은 F_1, F_2모두 통합에 영향을 줄 때를 의미한다. 따라서 아래와 같이 통합식을 구성할 수 있다.

$$Sim(d_1,Q) = Sim(\vec{C}_2,\vec{q}) * Sim(d_{21},\vec{q}) * (1-Sim(d_{11},\vec{q})) +$$
$$Sim(\vec{C}_2,\vec{q}) * Sim(d_{11},\vec{q}) * (1-Sim(d_{21},\vec{q})) + Sim(d_{11},\vec{q}) * Sim(d_{21},\vec{q})$$

7. 실 험

실험 목적은 비구조형 질의를 이용하여 텍스트 검색을 수행한 경우(UQ)와 사용자가 작성한 구조형 질의를 이용하여 구조 검색을 수행한 경우(SQ), 자동적으로 구조형 질의를 구성하여 구조 검색을 수행한 경우([8]에서 제안한 방법으로 VT로 명기한다), 마지막으로 본 논문에서 제안하는 구조형 질의를 이용하여 구조검색과 텍스트 검색을 병행하여 상호 보완하는 경우(HQ)의 검색 결과들을 평가하여 상호 비교하는 것이다. SQ는 SQ(AND)와 SQ(OR)로 분류하여 비교하였다. SQ(AND)는 복합 질의시에 AND연산으로서 결과를 얻은 것이며 SQ(OR)는 OR연산으로 검색하였을 경우 얻은 결과이다. 그리고 VT는 VT1과 VT1~VT5로 분류하여 비교하였는데 VT1은 가장 우선순위가 높은 구조형 질의에 대해서 구조검색을 수행하는 것이며, VT1~VT5는 상위 5개의 구조형 질의에 대해서 모두 검색을 수행하고 이에 대한 결과들을 통합하는 것이다. 측정 기준은 총 질의에 대한 평균정확율, 평균재현율, 평균 10-precision, 평균 F1이다. 10-precision은 상위 10개의 검색결과에 대해서 정확율을 계산한 것인데 일반적으로 사용자들은 상위 검색결과에 대해서 주목하고 하위 검색결과들은 크게 관심을 두지 않기 때문에 10-precision은 중요하다. F1은 정확율(Precision)과 재현율(Recall)을 하나의 값으로 나타내어 검색 모델의 전체적인 성능을 간단히 보여준다. F1은 2PR/(P+R) (P:정확율, R:재현율)로 정의된다.

본 실험에서 사용한 테스트 컬렉션은 Virginia Tech로부터 제공 받은CITIDEL 컬렉션이다. CITIDEL은 ACM을 비롯한 DBLP 등 다양한 과학분야의 저널에 대한 문서로 구성되어 있다. 이 중 ACM의 메타데이타로 구성된 98,000여건의 문서에서 초록(abstract)을 포함하고 있는 문서 39,698건을 실험 대상으로 하였다. 실험 대상이 되는 문서들은 모두 4개의 메타데이타 필드 - title, abstract, publication, author-로 구성되어있다. 적합문서 판단을 위한 질의는 단순질의와 복합질의의 두 가지 종류로 구성하였는데, 단순질의란 하나의 필드에 대해서만 질의를 구성하는 것이며 복합질의란 두 개의 메타데이타 필드를 이용하여 질의를 구성한 것이다. 본 실험에서는 적합문서 판단을 위한 두 가지 형태의 질의는 Virginia Tech에서 제공받은 것으로 같은 질의 형태 및 질의를 사용함으로써 결과 비교를 용이하게 하고자 하였다. 단, 질의에 해당하는 적합문서들은 제공받지 못하여 적합문서 판단은 ICU의 IRNLP 연구실의 학생들에 의해 수행되었고 따라서 Virginia Tech의 결과와 다소 상이할 수도 있다.

단순 질의는 하나의 메타데이타 필드에 대하여 질의를 하는 것으로서 title에 대한 질의, author에 대한 질의, publication에 대한 질의, 그리고 text에 대한 질의로 4종류로 분류할 수 있다. 즉 단순질의는 <title, author, publication, text> 중 하나로 구성한다.

복합질의는 두 개의 메타데이타 필드에 대하여 질의를 하는 것으로서 단순질의의 메타데이타 필드 중 두 개의 필드 조합으로 구성된다. 따라서 복합질의는 <title + author, title + publication, title + text, author + publication, author + text, publication + text> 중 하나로 구성되며 총 6종류로 구분된다.

단순질의는 각 종류별로 8개의 질의로 구성하였고 복합질의는 각 종류별로 5개의 질의로 구성하여 총 62개의 질의로서 실험을 수행하였다. 표 1은 각 질의형태에 따른 종류별 질의를 예로 보여준다.

위의 질의들을 이용하여 UQ, SQ(AND, OR), VT(VT1, VT1~VT5), 그리고 HQ에 대하여 검색을 수행하였으며 평균정확율, 재현율, 10-precision 그리고 F1으로 성능을 측정하였다. 성능 비교는 아래 테이블 1과 같다.

7.1 UQ, SQ, VT1의 결과 비교

결과에서 볼 수 있듯이, 구조검색(SQ)이 비구조 검색(UQ) 보다 우수하다. 이것은 비구조 검색(UQ)보다 구조검색(SQ)이 보다 사용자의 검색의도를 정확하게 반영하였기 때문이며 사용자의 의도에 맞지 않는 메타데이타 필드들에 대해서는 검색을 수행하지 않았기 때문이다. 구조 검색과 VT1은 그 검색 결과는 비슷하다. 이는 사용자가 직접 구조질의를 작성한 경우와 VT의 자동적으로 변환한 구조질의가 유사하기 때문이다.

7.2 SQ, VT1, VT1~VT5의 결과 비교

자동적으로 구조질의로 변환하여 상위 5개의 질의를 구성하여 검색하여 통합한 경우(VT1~VT5)는 SQ보다 다소 좋은 결과를 보인다. 왜냐하면 SQ나 VT1은 단 하나의 메타데이타 필드에 대해서만 검색을 수행하지만

표 1 실험 결과 비교

구분(개수)	종류 (개수)	예
단순질의 (32)	Title(8)	"image retrieval", "dynamic query"
	author(8)	"edward fox", "shneiderman"
	publication(8)	"toms", "jacm", "ACM SIGMOD"
	text(8)	"user profile", "natural language processing"
복합질의 (30)	Title+author(5)	"image retrieval" + "nascimento"
	Title+publication(5)	"image retrieval" + "tois"
	Title+text(5)	"image retrieval" + "relevance feedback"
	author+publication(5)	"susan brennan" + "tois"
	author+text(5)	"susan brenan" + "interface design"
	Text+ publication(5)	"user profile" + "tois"

테이블 1 실험 결과 비교

평균	UQ	SQ(AND)	SQ(OR)	VT1	VT1~VT5	HQ
정확율	26.05%	55.08%	50.10%	50.52%	51.05%	54.37%
재현율	40.72%	39.55%	79.87%	79.79%	80.62%	86.30%
10-Precision	28.24	65.87	72.50	80.08	82.01	84.12
F1	30.01	46.29	60.17	60.59	62.51	66.71

VT1~VT5는 적어도 두 개 이상의 메타데이타 필드를 검색하기 때문이다. 예를 들어 질의가 "information retrieval"일 경우 SQ나 VT1은 <(title, "information retrieval")>, <(text, "information retrieval")>, <(publication, "information retrieval")>, <(title, "information"), (text, "retrieval")>, <(text, "information"), (title, "retrieval")>, 등 다양하게 구성될 수 있는 구조형 질의 중 하나의 구조형 질의만을 선정하여 검색하게 되지만 VT1~VT5는 이러한 구조형 질의 중에서 상위 5개의 구조형 질의에 대해서 검색을 수행하여 결과를 얻기 때문에 사용자가 지정하지 않은 필드에서도 적합한 문서를 찾아낼 수 있다.

7.3 VT1~VT5 와 HQ 결과 비교

질의와 가장 적합한 필드를 검색하여 구조형 질의로 변환하여 검색하고 텍스트를 보완적으로 검색하여 통합한 방안(HQ)이 이들 중 가장 좋은 성능을 나타낸다. 이러한 이유는VT1~VT5는 다음과 같은 세 가지의 단점을 갖기 때문이다.

1) 필요 이상의 메타데이타 필드의 확장

VT1~VT5는 질의가 나타날 수 있는 모든 필드를 검색하고 그 중 상위 5개의 구조형 질의를 작성하므로 질의에 따라서 상위 5개의 구조형 질의라 하더라도 좋지 않은 구조형 질의가 발생할 수도 있다. 좋지 않은 질의란 구조형 질의의 메타데이타 필드를 검색하지 않았다면 더 좋은 결과를 얻을 수 있는 경우를 말한다. 예를 들어 질의가 "information retrieval"일 경우 상위 5개의 질의로서 <(publication, "information retrieval")>이 포함될 경우 이 구조형 질의에 대해서도 검색을 수행하고 결과를 통합하게 된다. Publication 필드를 검색하였을 경우 "information retrieval"에 관련된 적합한 문서를 찾을 가능성보다는 그렇지 못할 가능성이 더 크다. 왜냐하면 많은 publication들이 "information retrieval"과 관련이 없어도 "information"이라는 명칭을 가질 수 있기 때문이다.

2) 질의어의 분할

이미 상술하였듯이, VT에서는 사용자의 질의에 대하여 가능한 모든 구조형 질의를 구성하여 Bayesian network를 기반으로 메타데이타 필드와 질의간의 발생 확률을 계산하여 우선순위를 결정한다. 질의가 "information retrieval"일 경우 다음과 같은 상위 5개의 구조형 질의가 생성된다고 가정하자(Q1~Q5).

Q1 = <(title, "information retrieval")>
Q2 = <(text, "information retrieval")>
Q3 = <(title, "information"), (text, "retrieval")>
Q4 = <(title, "retrieval"), (text, "information")>
Q5 = <(publication, "information retrieval")>

VT는 상위 5개의 구조형 질의에 대하여 검색하고 결과를 통합한다. Q1, Q2와는 달리 Q3, Q4는 질의가 메타데이타 필드에 따라 분리되어 있다. 질의가 분리됨에 따라 "information system", "information processing", "information management" 등 "information retrieval"과는 관련이 없는 정보들이 검색될 수 있다. 이것은 사용자는 질의는 "information retrieval"임에도 불구하고 이를 분리하여 각 필드에 질의함에 따라 사용자의 정보

요구를 잘못 추론함으로써 좋지 않은 결과를 초래한 것으로 볼 수 있다.

3) 구조화 질의 순위에 따른 가중치 반영 부재

VT에서는 상위 5개의 질의의 우선순위를 정할 때 복잡한 Bayesian network 기법을 도입하였음에도 생성된 구조형 질의간의 순위를 결정할 때만 사용할 뿐 검색에 직접적으로 이용하지 않는다. 즉, Q1은 가장 높은 우선순위를 갖는 구조형 질의이며, Q5는 가장 낮은 우선순위를 갖는 구조형 질의이지만 Q1으로부터 얻은 결과와 Q5로부터 얻은 결과를 차별화하지 않고 있다. 다시 말하면 Q1으로부터 검색된 결과는 Q5로부터 검색된 결과보다 높은 가중치를 반영하여야 하지만 VT에서는 그러한 가중치 반영을 하지 않고 있다.

본 논문에서 제시하는 HQ는 2개의 필드 (시스템 지정 필드와 텍스트 필드) 혹은 3개의 필드 (사용자 지정 필드, 시스템 지정 필드, 그리고 텍스트 필드)만을 검색대상으로 하기 때문에 VT의 단점인 1) 불필요한 메타데이타 필드의 검색을 제한하였으며 가장 우선순위가 높은 메타데이타 필드를 검색대상으로 하고 텍스트를 보완적으로 검색하므로 위 예제의 경우 Q1과 Q2의 구조형 질의만을 검색하게 되며VT와는 달리 질의를 분할하여 검색하지 않는다. 그리고 메타데이타 필드와 질의간의 적합성을 가중치로 반영하기 때문에 우선순위 결정시 Q1으로부터 얻은 결과에 더 많은 가중치를 부여할 수 있다.

8. 결론 및 향후 연구방향

본 논문에서는 사용자가 메타데이타의 특성을 이해하지 못하여도 질의와 필드간의 적합성을 측정하여 사용자의 정보 요구를 추론함으로써 자동적으로 구조형 질의로 변환하는 방안을 제시하였으며 구조형 질의를 이용하여 구조 검색만 수행하는 것이 아니라 텍스트 검색을 병행하여 상호 보완하며, 검색된 결과들을 통합할 때 질의와 필드간의 적합성을 가중치로 활용하여 통합에 반영하는 방안을 제안하였다.

제안된 기법은 사용자가 메타데이타의 특성을 이해하지 못함에 따라 잘못된 구조질의를 하는 경우에도 사용자의 오류를 감안하여 적합한 결과를 제공해 줄 수 있다. 또한 사용자가 적합하게 구조질의를 작성하였을 경우에도 얻을 수 없는 결과를 다른 메타데이타 필드나 텍스트를 보완 검색함으로써 사용자가 원하는 결과를 제공할 수 있다.

실험을 통하여 구조검색이 비구조형 질의를 이용한 텍스트 검색보다 우수하다는 것을 보였으며 사용자의 구조형 질의뿐 아니라 시스템이 추론하여 구성한 시스템의 구조형 질의도 이용하여 구조검색하며, 텍스트를 보완적으로 검색하여 이들 결과들을 가중치를 부여하고 통합할 때 가장 우수하다는 것을 보였다.

구조형 질의를 자동적으로 구성할 때 중요한 것은 여러 메타데이타 필드 중, 어떤 필드를 검색 대상으로 결정하는 것이다. 너무 많은 필드를 검색하게 되면 효율이나 결과 면에서 좋지 않은 결과를 얻게 되며 너무 적은 필드를 검색하게 되면 적합한 결과들을 충분히 찾아낼 수가 없다. [8]에서는 상위 5개의 구조형 질의로 검색을 결정하였다. 상위 5개의 구조형 질의를 검색하는 것은 질의에 따라서 잘못된 구조질의가 상위로 랭크될 가능성이 있다. 즉, 너무 많은 필드를 검색대상으로 포함시킬 수가 있다. 이에 본 논문에서는 Bayesian network가 아닌 벡터 모델을 기반으로 간단히 구조질의로 전환하고 전환된 메타데이타 필드를 검색한 후 텍스트 필드를 보완적으로 검색하여 이 두 결과를 가중치를 반영하여 통합하였다.

다양한 필드가 있을 때 검색할 필드와 검색하지 않을 필드를 명확히 결정하고 가장 적합한 필드에서 도출된 결과들은 다른 필드에서 얻은 결과보다 차별화하여 순위를 결정할 때 사용자의 정보요구에 적합한 결과들을 효율적으로 찾아낼 수 있을 것이다.

적합한 메타데이타 필드와 텍스트 필드를 병행하여 검색한다고 하여도 찾을 수 없는 적합한 문서가 분명히 있다. 이를 모두 찾기 위해서는 전체 메타데이타 필드를 검색대상으로 하여야 하는데 이는 분명 득보다는 실이 많다. 또한 전체 메타데이타 필드를 검색대상으로 하여도 찾을 수 없는 적합한 문서가 있다. 예를 들어 질의가 "이순신"이라 하면 "이순신"에 대한 질의를 대상으로 문서를 검색할 뿐이다. 만일 "충무공"이라는 문서가 존재한다면 "이순신"이라는 질의로만은 적합한 문서로 판단하기 어렵다.

따라서 온톨로지[10]를 기반으로 하여 질의를 확장[4]한다면 보다 많은 적합한 문서를 얻을 수 있을 것이다. 온톨로지를 구축하여 검색할 경우 질의에 대한 적절한 확장이 가능해지고 또한 상호 이질적인 특성을 가진 메타데이타들을 동시에 검색할 수 있다. 차후 연구로서 온톨로지를 구성하여 기법에 도입하면 보다 월등한 결과를 얻을 수 있을 것으로 기대된다.

참 고 문 헌

[1] Baeza-Yates, R., Ribeiro-Neto, B.: Modern Information Retrieval. Addison Wesley, New York, NY (1999).

[2] Callan, J, P.: Document filtering with inference networks. In Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Zurich

Switzerland (1996) 262-269.
[3] Calado, P., Cristo, M., Moura, E., Ziviani B., Goncalves, M, A.: Combining link-based and content-based methods for web document classification. In Proceedings of the 12th International Conference on Information and Knowledge Management, New Orleans LA USA (2003) 394-401.
[4] Campos, L, M., Ferenandez-Luna, J, M., Huete, J, F.: Query Expansion in Information Retrieval Systems Using a Bayesian Network-Based Thesaurus. In Proceedings of the 14th Annual Conference on Uncertainty in Artificial Intelligence (UAI-98), San Francisco CA (1998) 53-60.
[5] Calado, P., Silva, A, S., Vieria, R, C., Laender, A, H, F., Ribeiro-Neto, B, A.: Searching Web Databases by Structuring Keyword-based Queries. In proceedings of the 11th International Conference on Information and Knowledge Management, McLean VA USA (2002) 26-33.
[6] Dumais, S, T., Platt, P., Hecherman, D., Sahami, M.: Inductive learning algorithms and representations for text categorization. In Proceedings of the 7th International Conference on Information and Knowledge Management CIKM'98, Bethesda Maryland USA (1998) 148-155.
[7] Deniman, D., Sumner, T., Davis L., Bhushan, S., Jackson.: Merging Metadata and Content-Based Retreival. In proceedings of Journal of Digital Information, Volume 4 Issue 3.
[8] Goncalves, M, A., Fox, E, A., Krowne, A., Calado, P., Laender, A, H, F., Silva, A, S., Ribeiro-Neto, B, A.: The effectiveness of Automatically Structured Queries in Digital libraries. In proceedings of the 2004 joint ACM/IEEE conference on Digital libraries - Volume 00, Tuscon AZ USA (2004).
[9] Haines, D., Croft, W, B.: Relevance feedback and inference networks. In Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Pittsburgh, PA, USA, June (1993) 2-11.
[10] S. H. Myaeng, D.-H. Jang, M.-S. Kim, and Z.-C. Zhoo. A flexible model for retrieval of SGML documents. In Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 138-145, Melbourne, Australia, August 1998.
[11] Passin, T, B.: Explorer's Guide to the Semantic Web, Manning press (2004).
[12] Ribeiro-Neto, B., Muntz, R.: A belief network model for IR. In proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Zurich, Switzerland, August (1996) 253-260.
[13] Silva, I., Ribeiro-Neto, B., Calado, P., Moura, E., Ziviani, N.: Linked-based and Content-Based Evidential Information in a Belief Network Model. In Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Athens Greece (2000) 96-103.
[14] Turtle, H, R., Croft, W, B.: Inference networks for document retrieval. In Proceedings of the 13th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Brussels, Belgium, September (1990) 1-24.
[15] Turtle, H, R., Croft, W, B.: Croft. Evaluation of an Inference network-Based Retrieval Model. ACM Transactions on Information Systems 9,3 (1991), 187-222.
[16] Valle, R, F., Ribeiro-Neto, B, A., Lima, L, R, S., Laender, A, H, F., Junior, H, R , F, F.: Improving text retrieval in medical collections through automatic categorization. In Proceedings of the 10th International Symposium on String Processing and Information Retrieval SPIRE 2003, Manaus Brazil (2003) 197-210.
[17] T. T. Chinenyanga and N. Kushmerick. Expressive retrieval from XML documents. In Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 163-171, New Orleans, Louisiana, USA, September 2001.
[18] N. Fuhr and K. Gross. XIRQL: a query language for information retrieval in XML documents. In Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 172-180, New Orleans, Louisiana, USA, September 2001.
[19] G. Navarro and R. Baeza- Yates. Proximal nodes: A model to query document databases by content and structure. ACM Transactions 15(4):400-435, Oct. 1997.

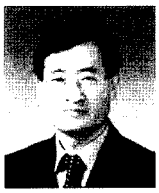
유 정 목
1996년 2월 충남대학교 자연과학대학 전산학과 이학사. 1998년 2월 충남대학교 자연과학대학 전산학과 이학석사. 2004년 2월 충남대학교 공과대학 컴퓨터과학과 이학박사 수료. 2005년 1월~현재 한국전자통신연구원 디지털홈연구단 인터넷 서버그룹 연구원

맹 성 현

1983년 미국 캘리포니아 주립대학 학사 1985년 미국 Southern Methodist University (SMU) 석사. 1987년 미국 Southern Methodist University (SMU) 박사. 1987년~1988년 미국 Temple University 교수. 1988년~1994년 미국 Syracuse University 교수 (tenured). 1994년~2003년 충남대학교 컴퓨터과학과 교수. 2003년~현재 한국정보통신대학교 교수

김 성 수

1997년 강원대학교 사범대학 영문학사 2005년 한국정보통신대학교 공학부 공학석사. 2007년 KT Biz컨설팅본부



이 만 호

1975년 2월 서울대학교 공과대학 응용수학과 공학사. 1977년 2월 한국과학기술원 전산학과 이학석사. 1991년 2월 미국 인디아나대학교 전산학 박사. 1980년 5월~1984년 8월 충남대학교 계산통계학과 조교수. 2000년 8월~2001년 8월 미국 Virginia Tech. 방문교수. 1991년 4월~현재 충남대학교 공과대학 전기정보통신공학부 컴퓨터전공 교수